



UNIVERSIDAD CARLOS III DE MADRID

Escuela Politécnica Superior

Departamento de Informática

TRABAJO FIN DE GRADO DE INGENIERÍA INFORMÁTICA

---

# Diseño y desarrollo de un portal eCommerce de ropa, calzado y accesorios deportivos

---

Autor:

Roberto Cañizares Sanz

Tutor:

Jesús Hernando Corrochano

Julio de 2016

*"El que lee mucho y anda mucho, ve mucho y sabe mucho."*

Miguel de Cervantes

# Agradecimientos

Parece que fue ayer cuando entré en la universidad, todavía lo recuerdo bien, un 5 de septiembre de 2011, ese día yo estaba lleno de nervios por saber cómo iban a ser las asignaturas, por conocer a mis nuevos compañeros pero sobre todo por dar el máximo de mí. Y es ahora cuando echas la vista atrás y te das cuenta de todo lo que lograste, de los buenos y malos momentos que pasaste, de las amistades y de las personas que siempre estuvieron cerca de mí animándome, compartiendo su tiempo e incluso ayudándome para lograr mis objetivos, es de justicia que yo ahora les dedique unas líneas.

Por ello, no puedo empezar sin agradecer a los pilares principales de mi vida que son mis padres, José y María Ángeles lo que soy, porque sin ellos no hubiera sido lo mismo. Les tengo que agradecer que desde pequeño me diesen una buena educación y me inculcaran grandes valores como la humildad, la paciencia, el esfuerzo y la responsabilidad. Aunque hemos pasado algún que otro momento complicado, vosotros no habéis dejado de apoyarme y creer en mí, eso merece todo mi respeto. No puedo olvidarme de otro pilar fundamental como es mi hermana Nuria con la que puedo contar siempre, ella lo es todo para mí. Gracias por todo vuestro tiempo, sacrificio, entrega y dedicación.

También quiero dar gracias a toda mi familia, por el enorme apoyo que he recibido todo este tiempo.

En el ámbito de la universidad, agradecer a todos aquellos compañeros de prácticas haberme aguantado con mis defectos y virtudes. Y como no, debo agradecer a mi tutor Jesús que se prestase para ayudarme en este trabajo en todo lo que le he necesitado. Muchas Gracias.

No puedo acabar los agradecimientos sin dedicarles unas líneas a Miguel, Javi, Víctor, Pedro, Alberto, Diego y Jose, amigos que son para toda la vida y que merece la pena acordarse de ellos en estos momentos tan importantes. Quiero hacer mención especial a dos de ellos que empezaron conmigo esta aventura: Miguel y Javi, en la que hemos pasado grandes ratos juntos y en la que nos hemos estado continuamente ayudando.

En estas últimas líneas quiero expresar lo orgulloso que estoy de toda esta gente y de poder tenerlos a mi lado. Me siento un privilegiado, porque es imposible encontrar gente que aúne tantos buenos valores como ellos.

A todos vosotros, Gracias.

## Prefacio

Si analizamos la situación actual podemos darnos cuenta que vivimos en un mundo cambiante y en continua evolución. El avance de las Tecnologías de la Información y Comunicación (TIC) han hecho que nuestra vida cotidiana sea producto de un cambio profundo, en el cual nos pasamos la mayor parte de nuestro tiempo conectados a la red, estemos donde estemos y con el dispositivo que sea. La tecnología se ha convertido en un factor fundamental de nuestro día a día, llegando casi al punto de que en ocasiones es totalmente transparente a nosotros y no somos capaces de concebir un mundo sin ella, ya que cada vez podemos realizar más acciones en torno a un mundo virtual, en el que no sólo navegamos, sino que también realizamos nuestras actividades diarias. Vivimos en un mundo "digital" pues es la era de "digitalización de nuestras vidas".

El comercio electrónico juega un papel importante en la vida de los internautas. La globalización en la red ha permitido la apertura de negocios en todo el mundo durante las 24 horas del día, los 7 días de la semana, así que ¿Por qué no crear un negocio e-commerce?

Esta idea es la que fomenta el desarrollo de este proyecto y veremos cómo llevarla a cabo para crear un negocio online de lo más proliferaante.

# Índice General

Agradecimientos.....	2
Prefacio.....	3
Índice General.....	4
Índice de Ilustraciones.....	10
Índice de Tablas.....	13
 Capítulo I – Motivación y Objetivos .....	16
Introducción .....	16
Motivación.....	16
Objetivos.....	17
Que se espera de este Proyecto .....	18
Medios empleados.....	18
Estructura de la memoria.....	19
 Capitulo II – Estado del Arte.....	20
Tecnologías.....	20
PHP ( <i>PHP: Hypertext Preprocesor</i> ) .....	20
.NET .....	21
JEE (Java Enterprise Edition) .....	22
Node.js .....	22
EJB (Enterprise Java Beans) .....	25
MySQL.....	28
Transformación Digital .....	30
Cloud Computing .....	30
Big Data .....	31
Movilidad .....	32
Puppet .....	32
Docker.....	33
CoreOS.....	34
Microservicios .....	34
Metodologías.....	36

Cascada .....	36
Ágil .....	37
Análisis del Negocio .....	39
Amazon .....	39
Zalando.....	40
El Corte Inglés.....	42
Under Armour .....	43
Capítulo III – Buscando una Solución Software .....	45
Idea Original .....	45
Restricciones Contractuales y Problemas Derivados .....	45
Selección de la Solución Software .....	46
Servidor de Aplicaciones .....	46
Arquitectura.....	47
Producto Final.....	50
Capítulo IV – Diseño del Sistema.....	51
Cliente .....	51
Estructura del Código .....	51
Prototipo de Aplicación .....	53
Usuario.....	54
Proveedor .....	58
Administrador .....	61
Servidor .....	64
Configuración del Servidor .....	64
Base de Datos .....	67
Base de Datos General.....	68
Banco BBDD .....	70
Capítulo V- Estrategia .....	73
Selección de la metodología a usar.....	73
Scrum, la metodología ágil más extendida .....	73

Equipo de Trabajo .....	74
Product Owner.....	74
Scrum Master .....	75
Equipo de desarrollo .....	75
Equipo de trabajo del proyecto .....	76
Modelo de trabajo .....	76
Eventos en Scrum.....	77
Artefactos en Scrum .....	79
Filosofía de trabajo .....	81
Reuniones presenciales del equipo de trabajo .....	81
Reuniones no presenciales del equipo de trabajo.....	81
Herramientas colaborativas.....	81
Repositorio de código fuente.....	82
Historias de Usuario .....	83
Usuario.....	85
Proveedor .....	92
Administrador .....	98
Pruebas.....	101
Matrices de Trazabilidad entre Historias de Usuarios .....	111
Usuario.....	111
Proveedor .....	112
Administrador .....	112
Matriz de Trazabilidad entre Historias de Usuarios y Pruebas.....	113
 Capítulo VI – Gestión del Proyecto .....	 115
Planificación.....	115
Presupuesto del Proyecto.....	116
Autor.....	116
Departamento.....	116
Descripción del Proyecto.....	117
Desglose del Presupuesto.....	117
Coste Personal .....	117

Coste Elementos Software y Hardware .....	118
Coste Total del Proyecto .....	119
Capítulo VII - Conclusiones .....	120
Conclusiones y Valoración Personal.....	120
Conclusiones Positivas.....	121
Conclusiones Negativas .....	121
Trabajos Futuros .....	122
Interfaz adaptativa .....	122
Redes Sociales .....	122
Añadir Comentarios y Valoraciones .....	122
Añadir Publicidad .....	123
Mejora de algunas funcionalidades .....	123
Recuperar la contraseña .....	123
Actualizar Stock.....	123
Swagger .....	123
Sistema de Logging (Trazas) .....	124
Capítulo VIII – Bibliografía.....	125
Bibliografía Referenciada en el Texto .....	125
Capitulo IX – Glosario de Términos .....	129
Anexo A – Manual de Usuario .....	131
Usuario.....	131
Registro de sesión.....	131
Inicio de sesión.....	132
Cierre de sesión.....	132
Acceder al perfil .....	133
Editar el perfil .....	133
Histórico .....	134
Acceder a productos .....	134
Acceder al producto .....	135



Añadir a suscripciones.....	135
Acceder a suscripciones.....	135
Añadir/Acceder al carrito.....	136
Realizar la compra.....	136
Proveedor.....	138
Registro de sesión.....	138
Inicio de sesión.....	139
Cierre de sesión.....	139
Acceder al perfil.....	140
Editar el perfil.....	140
Añadir a productos.....	141
Productos denegados.....	141
Acceder a productos.....	142
Acceder al producto.....	142
Editar un producto.....	143
Acceder a suscriptores.....	143
Facturación.....	143
Administrador.....	144
Inicio de sesión.....	144
Cierre de sesión.....	145
Validar los productos.....	145
Denegar los productos.....	145
Ver clientes.....	146
Editar cliente.....	146
Ver proveedores.....	147
Editar proveedor.....	147
Conciliación.....	148
Anexo B – Project Abstract.....	149
Motivation.....	149
Goals.....	150
Resume.....	151

Selected methodology .....	155
Scrum Events.....	156
Scrum Artifacts .....	157
Conclusions and Personal assessment .....	158
Positive Conclusions.....	158
Negative Conclusions .....	158
Future Works .....	159

# Índice de Ilustraciones

<i>Ilustración 1 - Arquitectura de una Plataforma .NET</i> .....	21
<i>Ilustración 2 - Código del lado del Servidor (Node.js)</i> .....	23
<i>Ilustración 3 - Código del lado del cliente (Jade)</i> .....	24
<i>Ilustración 4 - Despliegue del Servidor</i> .....	24
<i>Ilustración 5 - Despliegue de la Interfaz</i> .....	24
<i>Ilustración 6 - Creación de los EJB</i> .....	26
<i>Ilustración 7 - Creación de las interfaces</i> .....	26
<i>Ilustración 8 - Creación del Servlet</i> .....	27
<i>Ilustración 9 - Pedida del vale</i> .....	27
<i>Ilustración 10 - Obtención del vale</i> .....	27
<i>Ilustración 11 - MySQL</i> .....	28
<i>Ilustración 12 - Creación de una base de datos en MySQL</i> .....	29
<i>Ilustración 13 - Verificación por medio de la consola de comandos</i> .....	29
<i>Ilustración 14 - Cloud Computing</i> .....	30
<i>Ilustración 15 - Big Data</i> .....	31
<i>Ilustración 16 - Puppet</i> .....	32
<i>Ilustración 17 - Docker</i> .....	33
<i>Ilustración 18 - CoreOS</i> .....	34
<i>Ilustración 19 - Arquitectura Monolítica vs Microservicios</i> .....	35
<i>Ilustración 20 - Tienda Amazon</i> .....	39
<i>Ilustración 21 - Búsqueda de un producto en Amazon</i> .....	40
<i>Ilustración 22 - Tienda Zalando</i> .....	41
<i>Ilustración 23 - Búsqueda de un producto en Zalando</i> .....	41
<i>Ilustración 24 - Tienda del Corte Inglés</i> .....	42
<i>Ilustración 25 - Búsqueda de un producto en la tienda del Corte Inglés</i> .....	43
<i>Ilustración 26 - Tienda Under Armour</i> .....	44
<i>Ilustración 27 - Búsqueda de un producto Under Armour</i> .....	44
<i>Ilustración 28 - Funcionamiento de las tecnologías en la Aplicación Web</i> .....	48
<i>Ilustración 29 - Diagrama de Contexto</i> .....	49
<i>Ilustración 30 - Diagrama Contenedor Usuario</i> .....	49

<i>Ilustración 31 - Diagrama Contenedor Proveedor/Administrador</i> .....	50
<i>Ilustración 32 - Estructuración del Código</i> .....	51
<i>Ilustración 33 - Flujo Aplicación Web Usuario</i> .....	57
<i>Ilustración 34 - Flujo Aplicación Web Proveedor</i> .....	60
<i>Ilustración 35 - Flujo Aplicación Web Administrador</i> .....	63
<i>Ilustración 36 - JDBC Connection Pool</i> .....	65
<i>Ilustración 37 - Propiedades JDBC</i> .....	65
<i>Ilustración 38 - JDBC Resource</i> .....	66
<i>Ilustración 39 - JMS Connection Factory</i> .....	66
<i>Ilustración 40 - Base de Datos Aplicación Web</i> .....	72
<i>Ilustración 41 - Metodologías ágiles más usadas</i> .....	74
<i>Ilustración 42 - Ciclo de vida de la metodología Scrum</i> .....	76
<i>Ilustración 43 - Fechas de cada Sprints realizado en el proyecto</i> .....	77
<i>Ilustración 44 - Product Backlog del proyecto en Trello</i> .....	79
<i>Ilustración 45 - Ejemplo de Sprint en Trello</i> .....	80
<i>Ilustración 46 - Ejemplo de un Historia de Usuario en Trello</i> .....	80
<i>Ilustración 47 - Ciclo de vida del Sprint 1 en Trello</i> .....	81
<i>Ilustración 48 - Repositorio de código fuente en Dropbox</i> .....	82
<i>Ilustración 49 - Planificación del proyecto</i> .....	116
<i>Ilustración 50 - Visualización botón Crear cuenta Usuario</i> .....	131
<i>Ilustración 51 - Creación de un nuevo usuario</i> .....	131
<i>Ilustración 52 - Inicio de Sesión Usuario</i> .....	132
<i>Ilustración 53 - Cambio en la interfaz al iniciar de sesión en Usuario</i> .....	132
<i>Ilustración 54 - Cambio en la interfaz al cerrar sesión en Usuario</i> .....	132
<i>Ilustración 55 - Perfil Usuario</i> .....	133
<i>Ilustración 56 - Editar Perfil Usuario</i> .....	133
<i>Ilustración 57 - Histórico</i> .....	134
<i>Ilustración 58 - Menú productos</i> .....	134
<i>Ilustración 59 - Ejemplo productos mujer</i> .....	134
<i>Ilustración 60 - Información acerca de un producto</i> .....	135
<i>Ilustración 61 - Añadir/Cancelar Suscripción</i> .....	135
<i>Ilustración 62 - Suscripciones de productos</i> .....	136

<i>Ilustración 63 - Carrito Usuario .....</i>	<i>136</i>
<i>Ilustración 64 – Visualización de los datos de personales y envío .....</i>	<i>137</i>
<i>Ilustración 65 - Forma de Pago.....</i>	<i>137</i>
<i>Ilustración 66 - Confirmación de la compra.....</i>	<i>138</i>
<i>Ilustración 67 - Visualización botón Crear cuenta Proveedor.....</i>	<i>138</i>
<i>Ilustración 68 - Creación de un nuevo usuario.....</i>	<i>138</i>
<i>Ilustración 69 - Inicio de Sesión Proveedor .....</i>	<i>139</i>
<i>Ilustración 70 - Cambio en la interfaz al iniciar de sesión del Proveedor.....</i>	<i>139</i>
<i>Ilustración 71 - Cambio en la interfaz al cerrar sesión en Proveedor.....</i>	<i>139</i>
<i>Ilustración 72 - Perfil Proveedor .....</i>	<i>140</i>
<i>Ilustración 73 - Editar Perfil Proveedor.....</i>	<i>140</i>
<i>Ilustración 74- Añadir Producto Proveedor.....</i>	<i>141</i>
<i>Ilustración 75 - Productos denegados.....</i>	<i>141</i>
<i>Ilustración 76 - Productos Proveedor.....</i>	<i>142</i>
<i>Ilustración 77 - Producto Proveedor.....</i>	<i>142</i>
<i>Ilustración 78 - Editar Producto .....</i>	<i>143</i>
<i>Ilustración 79 - Suscriptor a un producto .....</i>	<i>143</i>
<i>Ilustración 80 - Facturación Proveedor.....</i>	<i>144</i>
<i>Ilustración 81 - Inicio de Sesión Administrador.....</i>	<i>144</i>
<i>Ilustración 82 - Cambio en la interfaz al iniciar de sesión del Administrador.....</i>	<i>144</i>
<i>Ilustración 83 - Cambio en la interfaz al cerrar sesión en Administrador .....</i>	<i>145</i>
<i>Ilustración 84 - Validación de productos .....</i>	<i>145</i>
<i>Ilustración 85 - Denegar producto de un proveedor.....</i>	<i>146</i>
<i>Ilustración 86 - Ver clientes en la zona de administración.....</i>	<i>146</i>
<i>Ilustración 87 - Edición de los clientes en la zona de administración .....</i>	<i>147</i>
<i>Ilustración 88 - Ver roveedores en la zona de administración.....</i>	<i>147</i>
<i>Ilustración 89 - Edición de los proveedores en la zona de administración .....</i>	<i>148</i>
<i>Ilustración 90 - Conciliación.....</i>	<i>148</i>
<i>Illustration 91 - Context Diagram .....</i>	<i>153</i>
<i>Illustration 92 – User Container Diagram .....</i>	<i>153</i>
<i>Illustration 93 – Supplier/Administrator Container Diagram.....</i>	<i>154</i>
<i>Illustration 94 - Life cycle of Scrum methodology .....</i>	<i>156</i>

## Índice de Tablas

<i>Tabla 1 - Historia de Usuario 1. UH-U-01.....</i>	<i>85</i>
<i>Tabla 2 - Historia de Usuario 2. UH-U-02.....</i>	<i>85</i>
<i>Tabla 3 - Historia de Usuario 3. UH-U-03.....</i>	<i>86</i>
<i>Tabla 4 - Historia de Usuario 4. UH-U-04.....</i>	<i>86</i>
<i>Tabla 5 - Historia de Usuario 5. UH-U-05.....</i>	<i>87</i>
<i>Tabla 6 - Historia de Usuario 6. UH-U-06.....</i>	<i>87</i>
<i>Tabla 7 - Historia de Usuario 7. UH-U-07.....</i>	<i>88</i>
<i>Tabla 8 - Historia de Usuario 8. UH-U-08.....</i>	<i>88</i>
<i>Tabla 9 - Historia de Usuario 9. UH-U-09.....</i>	<i>89</i>
<i>Tabla 10 - Historia de Usuario 10. UH-U-10.....</i>	<i>89</i>
<i>Tabla 11 - Historia de Usuario 11. UH-U-11.....</i>	<i>90</i>
<i>Tabla 12 - Historia de Usuario 12. UH-U-12.....</i>	<i>90</i>
<i>Tabla 13 - Historia de Usuario 13. UH-U-13.....</i>	<i>91</i>
<i>Tabla 14 - Historia de Usuario 14. UH-U-14.....</i>	<i>91</i>
<i>Tabla 15 - Historia de Usuario 14. UH-P-01.....</i>	<i>92</i>
<i>Tabla 16 - Historia de Usuario 15. UH-P-02.....</i>	<i>92</i>
<i>Tabla 17 - Historia de Usuario 16. UH-P-03.....</i>	<i>93</i>
<i>Tabla 18 - Historia de Usuario 17. UH-P-04.....</i>	<i>93</i>
<i>Tabla 19 - Historia de Usuario 18. UH-P-05.....</i>	<i>94</i>
<i>Tabla 20 - Historia de Usuario 19. UH-P-06.....</i>	<i>94</i>
<i>Tabla 21 - Historia de Usuario 20. UH-P-07.....</i>	<i>95</i>
<i>Tabla 22 - Historia de Usuario 21. UH-P-08.....</i>	<i>95</i>
<i>Tabla 23 - Historia de Usuario 22. UH-P-09.....</i>	<i>96</i>
<i>Tabla 24 - Historia de Usuario 24. UH-P-10.....</i>	<i>96</i>
<i>Tabla 25 - Historia de Usuario 25. UH-P-11.....</i>	<i>97</i>
<i>Tabla 26 - Historia de Usuario 26. UH-P-12.....</i>	<i>97</i>
<i>Tabla 27 - Historia de Usuario 27. UH-A-01.....</i>	<i>98</i>
<i>Tabla 28 - Historia de Usuario 28. UH-A-02.....</i>	<i>98</i>
<i>Tabla 29 - Historia de Usuario 29. UH-A-03.....</i>	<i>99</i>
<i>Tabla 30 - Historia de Usuario 30. UH-A-04.....</i>	<i>99</i>

Tabla 31 - Historia de Usuario 31. UH-A-05.....	100
Tabla 32 - Historia de Usuario 32. UH-A-06.....	100
Tabla 33 - Historia de Usuario 33. UH-A-07.....	101
Tabla 34 - Prueba de Aceptación. TEST-01.....	102
Tabla 35 - Prueba de Aceptación. TEST-02.....	102
Tabla 36 - Prueba de Aceptación. TEST-03.....	102
Tabla 37 - Prueba de Aceptación. TEST-04.....	103
Tabla 38 - Prueba de Aceptación. TEST-05.....	103
Tabla 39 - Prueba de Aceptación. TEST-06.....	103
Tabla 40 - Prueba de Aceptación. TEST-07.....	104
Tabla 41 - Prueba de Aceptación. TEST-08.....	104
Tabla 42 - Prueba de Aceptación. TEST-09.....	104
Tabla 43 - Prueba de Aceptación. TEST-10.....	105
Tabla 44 - Prueba de Aceptación. TEST-11.....	105
Tabla 45 - Prueba de Aceptación. TEST-12.....	105
Tabla 46 - Prueba de Aceptación. TEST-13.....	106
Tabla 47 - Prueba de Aceptación. TEST-14.....	106
Tabla 48 - Prueba de Aceptación. TEST-15.....	107
Tabla 49 - Prueba de Aceptación. TEST-16.....	107
Tabla 50 - Prueba de Aceptación. TEST-17.....	107
Tabla 51 - Prueba de Aceptación. TEST-18.....	108
Tabla 52 - Prueba de Aceptación. TEST-19.....	108
Tabla 53 - Prueba de Aceptación. TEST-20.....	108
Tabla 54 - Prueba de Aceptación. TEST-21.....	109
Tabla 55 - Prueba de Aceptación. TEST-22.....	109
Tabla 56 - Prueba de Aceptación. TEST-23.....	109
Tabla 57 - Prueba de Aceptación. TEST-24.....	110
Tabla 58 - Prueba de Aceptación. TEST-25.....	110
Tabla 59 - Prueba de Aceptación. TEST-26.....	110
Tabla 60 - Prueba de Aceptación. TEST-27.....	111
Tabla 61 - Trazabilidad entre Historias de Usuario e Historias de Usuario (Usuario) .....	111
Tabla 62 - Trazabilidad entre Historias de Usuario e Historias de Usuario (Proveedor).....	112

<i>Tabla 63 - Trazabilidad entre Historias de Usuario e Historias de Usuario (Administrador) .....</i>	<i>112</i>
<i>Tabla 64 - Matriz de trazabilidad entre Historias de Usuarios y Pruebas.....</i>	<i>114</i>
<i>Tabla 65 - Coste personal .....</i>	<i>118</i>
<i>Tabla 66 - Coste elementos software y hardware .....</i>	<i>118</i>
<i>Tabla 67 - Precio total del Proyecto .....</i>	<i>119</i>



# Capítulo I – Motivación y Objetivos

En este primer capítulo se detallan los motivos que me han impulsado a desarrollar este proyecto, los objetivos que ha de cumplir la solución software y lo que se espera de él. Además se detallan los medios utilizados y la estructura del documento.

## Introducción

La elección de crear un negocio e-commerce como tema principal de este proyecto se debe, en gran parte, al uso que hacemos de internet, porque gracias a él muchos consumidores utilizan sus smartphone, tablets, ordenadores o las propias televisiones para comprar o vender productos, creando un nuevo paradigma de intercambio de bienes. Los negocios e-Commerce tiene una importancia capital, ya que diariamente se realizan múltiples operaciones y compras electrónicas. Han tenido un crecimiento exponencial en muy pocos años y gran parte de ese éxito se debe a que han mejorado la prestación de sus servicios y han conseguido acercar al usuario la tienda tradicional, haciendo sus páginas web más accesibles.

¿Por qué son más beneficios los negocios e-commerce? Pues bien, son más beneficiosos porque la mayoría de sus consumidores agradecen la comodidad de poder realizar transacciones o compras sin tener que trasladarse físicamente a la tienda, de este modo se ahorran tiempo, energía y dinero.

Se pretende, por tanto, crear y potenciar un negocio e-commerce haciendo uso de las mejores tecnologías, las cuales se verán más adelante. Del mismo modo, se busca afianzar, ampliar y profundizar en los conocimientos y competencias que se han ido adquiriendo durante la etapa universitaria.

## Motivación

El presente proyecto persigue afianzar los conocimientos y habilidades que se han adquirido a lo largo del Grado de Ingeniería Informática. Combinar todo lo que he aprendido desde que entré en la universidad en 2011 hasta ahora en un único proyecto, es una tarea ardua y compleja que entraña múltiples dificultades, pero que a su vez es realmente motivadora porque culmina una maravillosa etapa universitaria.

La puesta en común de los conocimientos, técnicas y habilidades aprendidas en cada una de las siguientes asignaturas: programación, técnicas de búsqueda y uso de la información, técnicas de expresión oral y escrita, estructura de datos y algoritmos, fundamentos de gestión empresarial,

ficheros y bases de datos, principios de desarrollo software, ingeniería del software, interfaces de usuario y tecnologías informáticas para la web, son la motivación suficiente para desarrollar un proyecto ambicioso.

Otro de los motivos que me impulsó a desarrollar este proyecto está relacionado con el auge y la popularidad que está teniendo el comercio electrónico en el mundo laboral. Ya sabemos que internet es una fuente de oportunidades pero, eso sí, no todo vale y para montar un buen negocio online necesitas acercarte a la gente para conocer su opinión y como mejorar día a día.

Embarcarse en este gran proyecto tan interesante y ambicioso implica conocer de cerca las preocupaciones y problemas del comercio electrónico, y así poder aprender a analizarlos y subsanarlos de la mejor manera posible. Esta implicación puede ser una buena forma de acercarme al mundo empresarial que hay detrás de una carrera universitaria.

## Objetivos

El objetivo principal del proyecto consistirá en el desarrollo de una aplicación de comercio electrónico que facilite la operativa de una tienda de venta de ropa, calzado y accesorios deportivos. La tienda dispondrá de un amplio catálogo de productos, que son puestos a disposición de los usuarios por los proveedores. Los usuarios podrán acceder a la información detallada de cada producto, se podrán suscribir a él o incluso comprarlo si así lo desean. Para estos últimos pasos el usuario se debe registrar o iniciar sesión. Habrá también un portal de administración que será el encargado de que la tienda funcione a la perfección y de vigilar a usuarios y proveedores para evitar un mal uso de la aplicación. El sistema para el cobro de los productos se realizará por medio de una simulación de pasarelas de pago.

Los principales objetivos que componen este proyecto son:

- ~ Construir una aplicación eCommerce atractiva, útil y de fácil aprendizaje.
- ~ Construir diferentes portales para la gestión de la aplicación eCommerce.
- ~ Simular el sistema de cobro por medio de las pasarelas de pago.
- ~ Realizar un plan de pruebas para comprobar el correcto funcionamiento de la aplicación.

## Que se espera de este Proyecto

Como profesional potencial espero poder plasmar en éste proyecto todos los conocimientos adquiridos durante estos años en la universidad y que a la vez sirva como entrenamiento para poder acceder al mundo laboral.

El estudio e investigación de nuevas tecnologías emergentes está a la orden del día. Además de ampliar el abanico de conocimiento de los profesionales del sector, estos se pueden ver beneficiados con la explosiva y creciente creatividad de tecnologías, ya que pueden dar a sus negocios un aspecto que no han tenido en cuenta otras empresas.

La realización de este proyecto se enmarca como un punto de inicio en esta larga carrera profesional, ya que comenzar desde cero un “negocio virtual” no es nada sencillo, se requiere de conocimientos sobre el área, conversaciones con el cliente, un buen diseño de la página web y de la base de datos, una serie de requisitos y diferentes pruebas que garanticen la eficacia y eficiencia de nuestro “negocio virtual” para que pueda competir en el mercado con las diferentes empresas del mismo sector.

## Medios empleados

En este apartado se van a documentar los diferentes medios que se han utilizado para el desarrollo del proyecto.

- ~ Los *Elementos hardware* son aquellos componentes físicos que se han usado para realizar el proyecto.
  - PC portátil HP Pavilion 15-p265ns: Ordenador con el que se ha trabajado para desarrollar la aplicación eCommerce. Sus características básicas son las siguientes:
    - Sistema Operativo: Microsoft Windows 8.1
    - Procesador: AMD A10-7300 Radeon R6 de 1.90 Ghz
    - Memoria RAM: 16 GB
- ~ Los *Elementos software* son todas aquellas herramientas lógicas que han hecho posible la elección y elaboración del proyecto.
  - Eclipse Luna for Java Developer 4.4.0 [1]
  - Google Chrome 48 [2]
  - MySQL Workbench 6.3 [3]
  - SoapUI 5.2.1 [4]
  - Selenium 2.52 [5]

- Insomnia REST Client [6]

## Estructura de la memoria

El documento queda constituido por nueve capítulos (en los que se detalla el desarrollo del proyecto, la bibliografía consultada y el glosario de términos). Más detalladamente:

- ~ *Capítulo I – Motivación y Objetivos:* breve capítulo que tiene como finalidad presentar el proyecto, describir los motivos que impulsan a realizarlo, explicar los objetivos, lo que se espera de él y los medios empleados.
- ~ *Capítulo II – Estado del arte:* se exponen las posibles tecnologías aplicables a este trabajo, las metodologías de desarrollo más extendidas y el análisis de los negocios eCommerce del mercado actual.
- ~ *Capítulo III – Buscando una solución software:* su finalidad es exponer la idea original del proyecto y analizar las diversas arquitecturas para el desarrollo de la aplicación, así como del servidor de aplicaciones.
- ~ *Capítulo IV – Diseño del Sistema:* aborda temas tan importante como la estructuración del código, el diseño de la base de datos, los prototipos de la aplicación y el flujo entre las pantallas, y la configuración del servidor.
- ~ *Capítulo V – Estrategia:* describe la metodología a seguir, el equipo de trabajo, las historias de usuario que el cliente desea para su aplicación y el sistema de pruebas para verificar el correcto funcionamiento de todo el conjunto.
- ~ *Capítulo VI – Gestión del Proyecto:* se expondrá la planificación final que se ha llevado a cabo para desarrollar el sistema y se establecerá el presupuesto del proyecto.
- ~ *Capítulo VII – Conclusiones:* se añaden las conclusiones extraídas a la hora de trabajar en el proyecto y se combinan con una pequeña valoración personal del mismo. También se deja la puerta abierta a trabajos futuros y a las posibles mejoras del sistema.
- ~ *Capítulo VIII – Bibliografía:* donde se exponen las referencias utilizadas durante el desarrollo de todo el proyecto.
- ~ *Capítulo IX – Glosario de Términos:* donde se exponen los términos utilizados durante todo el desarrollo del proyecto.
- ~ *Anexo A – Manual de Usuario:* muestra un manual de usuario para ayudar ante las posibles dudas que puedan surgir con el uso de la aplicación.
- ~ *Anexo B – Project Abstract:* breve resumen en inglés del presente documento.

## Capítulo II – Estado del Arte

A lo largo de este capítulo se realizará una introducción de las tecnologías informáticas actuales que guardan relación con el contenido del proyecto o que afectan directamente al mismo. Por lo tanto, se pretende realizar un estudio de las posibilidades que nos ofrece el mercado para desarrollar un proyecto de estas características.

El análisis del estado del arte es necesario para poder tomar la decisión más acertada a la hora de decidir cómo se va a desarrollar el proyecto. Se tendrá en cuenta las metodologías y los puntos clave del análisis del mercado actual.

### Tecnologías

A lo largo de este apartado haremos un estudio de los diversos tipos de tecnologías que se pueden usar en el ámbito del comercio electrónico.

#### *PHP (PHP: Hypertext Preprocesor)*

Desde su nacimiento a mediados de los años 90, ha recorrido un largo camino hasta ser uno de los lenguajes más importantes de programación web dinámica. PHP es capaz de realizar determinadas acciones de una forma fácil y eficaz sin tener que generar programas en un lenguaje distinto al de HTML. El sistema fue originalmente diseñado en Perl, en el año 1994 por Rasmus Lerdorf, como un conjunto simple de ficheros binarios Common Gateway Interface (CGI) escritos en el lenguaje de programación C, que permitía la interpretación de un número limitado de comandos.

El sistema fue denominado Personal Home Page Tools y adquirió relativo éxito gracias a que otras personas pidieron a Rasmus que les permitiese utilizar el programa en sus propias páginas. Dada la aceptación del primer PHP y de manera adicional, Rasmus tuvo la necesidad de crear páginas dinámicas que trabajasen con formularios, creó una serie de etiquetas a las que denominó "Form Interpreter", y lo sacó al público con el nombre PHP/FI en 1995[7].

La siguiente gran contribución al lenguaje PHP fue a mediados de 1997 cuando se incluyeron nuevas funcionalidades como el soporte a protocolos de internet y el soporte a la gran mayoría de las bases de datos comerciales. Todas estas mejoras sentaron las bases de la versión 3 de PHP [8].

La versión 4 de PHP utiliza el motor Zend, desarrollado para cubrir las necesidades actuales y solucionar algunos inconvenientes de la versión 3. Las mejoras de esta versión se basaban en su mayor independencia del servidor web y en su rapidez, ya que primero se compila y luego se ejecuta, mientras que antes se ejecutaba a la misma vez que se interpretaba el código.

La última versión de PHP es la 5 que utiliza el motor Zend-2 y presenta mejoras significativas y un entorno de programación orientado a objetos mucho más completo, que permite que PHP proporcione un alto rendimiento a las aplicaciones Web empresariales a nivel de plataformas J2EE y .NET [9].

Una nota importante de PHP es que ha sido desarrollado inicialmente para entornos UNIX y es en este sistema operativo donde se aprovecha mejor sus prestaciones y se consigue un mayor rendimiento.

## .NET

La plataforma .NET nace a finales del año 2000 para hacer frente al éxito que estaba cosechando Java de Sun Microsystems. .NET no es más que un conjunto de tecnologías para desarrollar y utilizar componentes que nos permitan crear formularios web, servicios web y aplicaciones para Windows [10].

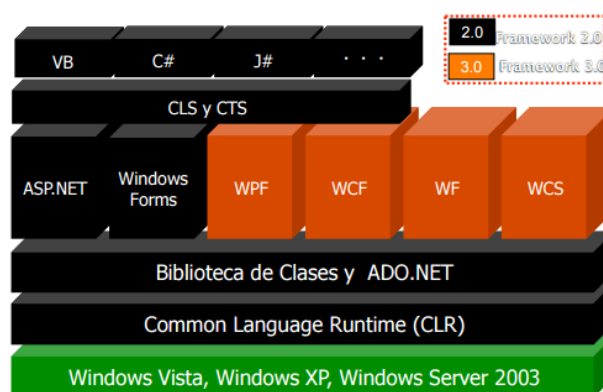


Ilustración 1 - Arquitectura de una Plataforma .NET

.NET ofrece una plataforma sencilla y potente para distribuir el software en forma de servicios que puedan ser suministrados remotamente y que puedan comunicarse y combinarse unos con otros de manera totalmente independiente de la plataforma, lenguaje de programación y modelo de componentes con los que hayan sido desarrollados [11].

Esta plataforma ofrece una manera rápida y económica pero a la vez segura y robusta de desarrollar aplicaciones, permitiendo una integración más rápida y ágil entre empresas. Además de un acceso más simple y universal a todo tipo de informaciones desde cualquier tipo de dispositivos.

El “framework” o marco de trabajo, constituye la base de la plataforma .NET y denota la infraestructura sobre la cual se reúnen un conjunto de lenguajes, herramientas y servicios que simplifican el desarrollo de aplicaciones en entornos de ejecución distribuidos [12].

## JEE (Java Enterprise Edition)

Es una especificación de especificaciones, es decir, una guía de pautas que rige el desarrollo de una aplicación Java, por consiguiente no es un producto concreto. JEE es una especificación que define [13]:

- ~ JSR (*Java Specification Request*) se crea y se presenta para su aprobación cuando una persona o entidad cree que es necesario la presencia de una determinada tecnología dentro de las plataformas basadas en Java. Si esta petición es aprobada entonces se crea una especificación, un documento en el cual se describe dicha tecnología, sus partes, las relaciones entre las mismas y los roles de las personas que usarán dicha tecnología. Además ha de proporcionar un test de compatibilidad y una implementación de referencia.
- ~ JCP (*Java Community Process*), es un organismo formado por empresas, asociaciones y particulares cuyo objetivo es asegurar la evolución de las plataformas Java.
- ~ CTS (*Compatibility Test Suite*), es un test de compatibilidad que consta de una serie de pruebas, más de 15000, cuyo objetivo es asegurar que un producto, típicamente un servidor de aplicaciones, está conforme con la especificación de J2EE. Si pasa el test de compatibilidad estamos seguros de que un producto cumple con todas las normas que marca la especificación.
- ~ Los J2EE BluePrints son un conjunto de guías de desarrollo y de prácticas, que podemos descargar desde la web de Sun Microsystems, que definen patrones de diseño y prácticas recomendadas en la creación de aplicaciones J2EE.

## Node.js

Fue creado por Ryan Dahl y presentado por primera vez en el JSConf2009, su evolución está apadrinada por la empresa Joyent y en poco tiempo ha logrado desplazar en popularidad a otras implementaciones de server-side javascript como Rhino, Ringo y Narwhal.

Node.js está pensado para facilitar el desarrollo de aplicaciones basadas en red, rápidas y fiables. Utiliza un modelo I/O (entrada/salida) orientado a eventos y basado en el ‘no-

bloqueo', que lo hace ligero y eficiente, ideal para aplicaciones en tiempo real que hagan uso de datos intensivos y que se ejecuten a través de dispositivos distribuidos.

Node.js corre sobre V8, el motor de Javascript de Google Chrome, el cual es conocido por su velocidad. Por otro lado, Mozilla está implementando Node.js sobre su propio motor, con lo cual esto generará competencia y seguramente la mejora de estos motores en los próximos meses [14].

Node.js es una plataforma bastante flexible en cuanto a los protocolos que puede manejar (TCP, DNS y HTTP). Además está especialmente indicado para desarrollar aplicaciones web, aplicaciones de red, aplicaciones para líneas de comandos, juegos online, chats, redes sociales, herramientas de traducción en tiempo real y mucho más [15].

A continuación, se muestra un laboratorio el cual hace uso de esta tecnología. En el lado del servidor se emplea Node.js, que permite una creación más rápida de servidores web, y en el lado del cliente se emplea el template Jade que proporcionara la interfaz de un chat. Antes de poder desplegar el servidor se debe instalar las herramientas necesarias.

```
var http = require('http');
var jade = require('jade');
var express = require('express');
var app = express();
var PORT = 3000;
var IP = '192.168.1.37';

app.set('view engine', 'jade');

app.get('/', function(req, res) {
    res.render('index');
});

var server = app.listen(PORT, IP, function () {
    var host = server.address().address;
    var port = server.address().port;

    console.log('El servidor esta escuchando en http://%s:%s', host, port);
});
```

Ilustración 2 - Código del lado del Servidor (Node.js)



```

doctype html
html
  head
    meta(charset='utf-8')
    meta(name='viewport', content='user-scalable=no, initial-scale=1, maximum-scale=1, minimum-scale=1')

    title Chat con Nodejs + socket.io
    style
      include ./css/style.css
      include ./css/styleMovil.css
      include ./css/styleTablet.css
    script
      include ./js/script.js
  body
    #chat
      form#send-messages(action='')
        select#channel(name='channel')
          option(value='A') Sala A
          option(value='B') Sala B
        input#message(type='text', placeholder='Escribe tu mensaje...')
        input#btn1(type='submit', value='Enviar')
      section#messages
        h3#flip Usuarios Conectados
        #nicknames

```

Ilustración 3 - Código del lado del cliente (Jade)

Por último, se podrá observar el resultado de la ejecución:

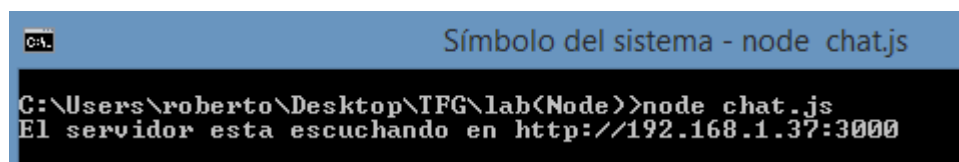


Ilustración 4 - Despliegue del Servidor



Ilustración 5 - Despliegue de la Interfaz

## EJB (Enterprise Java Beans)

Enterprise Java Beans es una arquitectura que define la forma de construir componentes distribuidos del lado del servidor, garantizando que los componentes programados sean escalables, eficaces y seguros a pesar de lo sencillo de su desarrollo.

Están diseñados para encapsular la lógica de negocio y proteger al desarrollador de aplicaciones para que no se tenga que preocupar por aspectos de nivel de sistema, tales como transacciones, seguridad, concurrencia, gestión de recursos, persistencia, manejo de errores e independencia del sistema operativo.

Los EJB's 3.0 hacen uso de las anotaciones que prescinden de elementos en ocasiones engorrosos como los descriptores. Es posible obtener la referencia de una instancia de un Enterprise Bean mediante inyección de dependencias a través de anotaciones. Esta inyección facilita la creación de las instancias de objetos, separando la definición de una clase de la instanciación de los objetos de los que depende. Otra ventaja es la sencilla forma de declarar la persistencia de los Entities, siendo JPA la API de persistencia [16]. Actualmente el estándar define los siguientes tipos de EJB:

- ~ Los *Session Bean* encapsulan la lógica de negocio que un cliente invoca programáticamente de manera local, remota o a través de un servicio web. Existen tres subtipos:
  - *Stateful*: Almacena el estado entre las diferentes invocaciones a sus métodos.
  - *Stateless*: No se mantiene ningún estado conversacional con el cliente, cuando alguien invoca un método de una *Stateless Bean* se guarda un estado pero sólo en la duración de la invocación, cuando el método termina el estado no se retiene. Excepto en la invocación de métodos donde cada llamada es independiente de las demás.
  - *Singleton*: Es instanciada una única vez por cada aplicación y su ciclo de vida corresponde con el de la aplicación misma. Están pensados para circunstancias donde una única instancia de Enterprise Bean es compartida y accedida concurrentemente por todos los clientes. Mantienen el estado entre invocaciones de clientes pero pueden no mantenerlo ante caídas o apagados de servidor.

- ~ Los *Message-Driven* son Enterprises Beans que permiten que aplicaciones Java EE procesen mensajes asincrónicamente. Estos beans actúan normalmente como JMS listeners, que son similares a los listener de eventos, pero en vez de recibir eventos reciben mensajes.
- ~ Los *Beans de entidad* quedaron obsoletos en EJB 3.1 y opcionales en EJB 3.2.

A continuación, se puede observar un laboratorio donde se pone en práctica esta tecnología. La funcionalidad es simple, el cliente selecciona en la interfaz el tipo de código que quiere (vale, operación o código) y los EJB trabajan por detrás para ofrecerle el código, que visualizará en la siguiente interfaz.

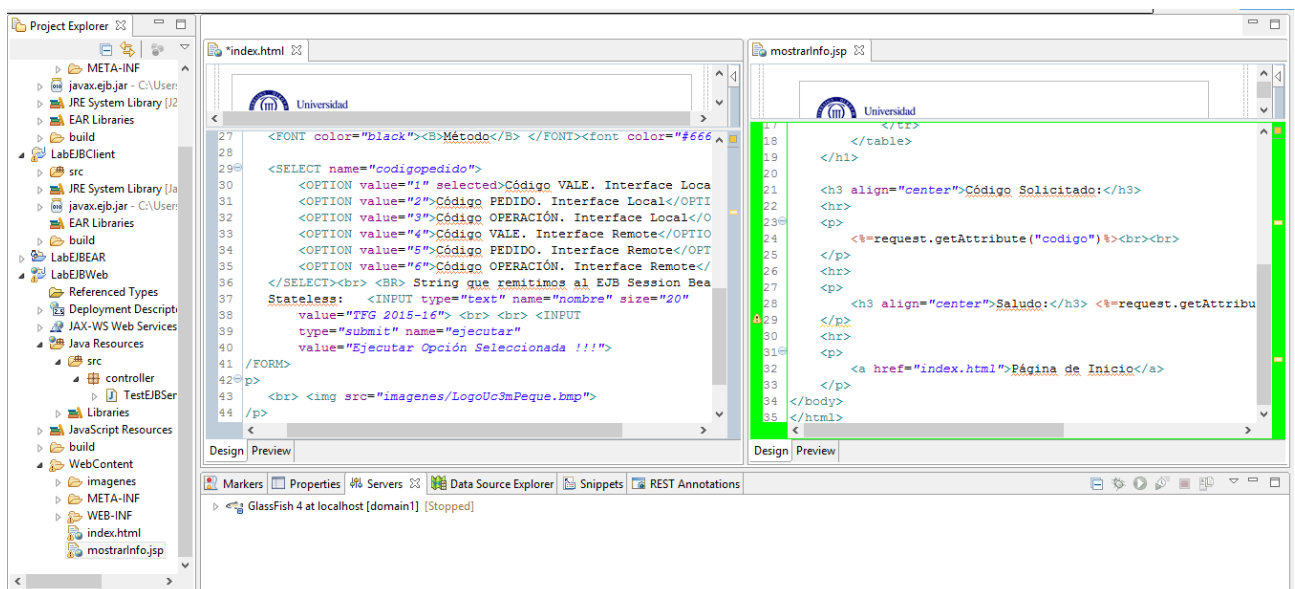


Ilustración 6 - Creación de los EJB

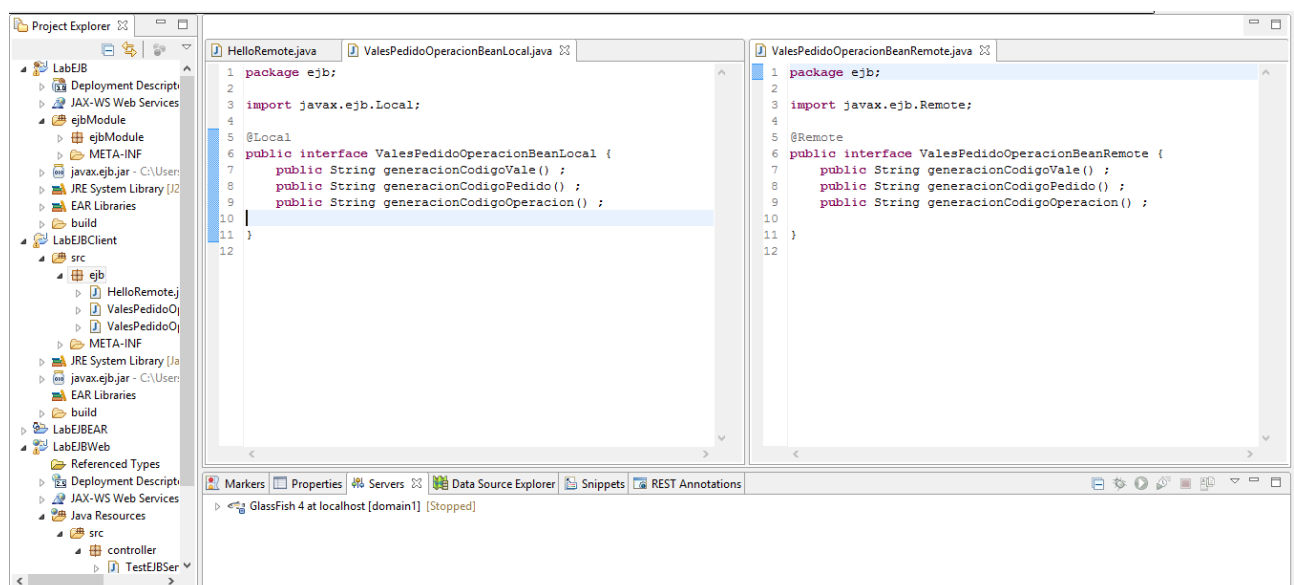


Ilustración 7 - Creación de las interfaces

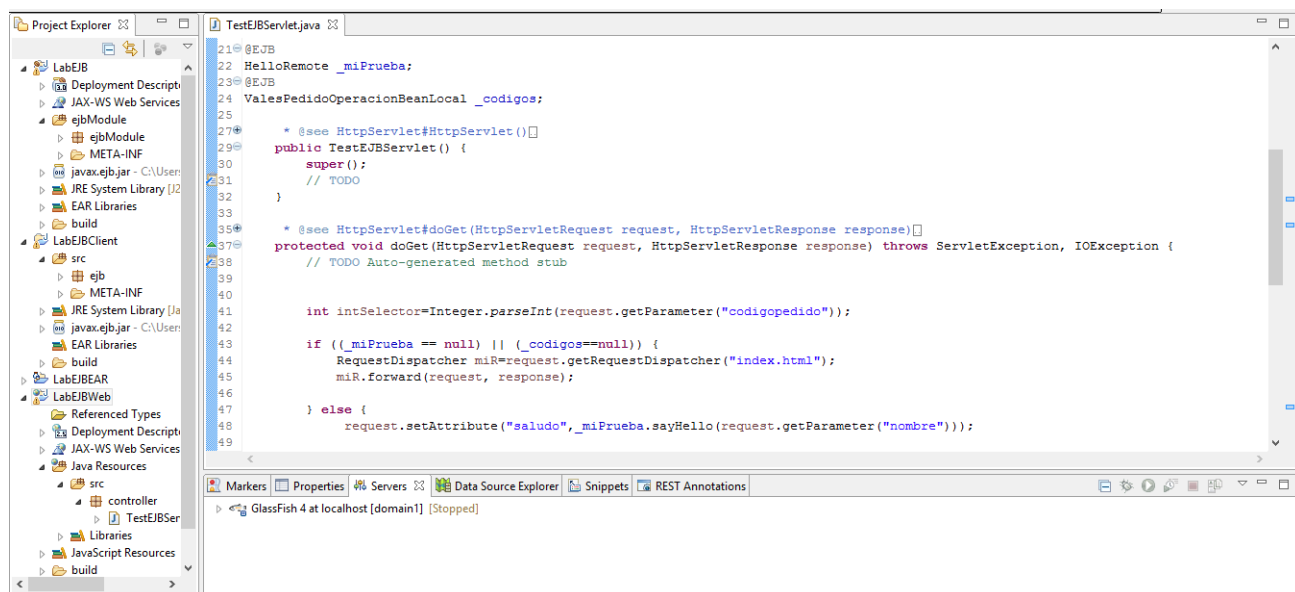


Ilustración 8 - Creación del Servlet

El resultado del laboratorio se puede ver en las siguientes ilustraciones:

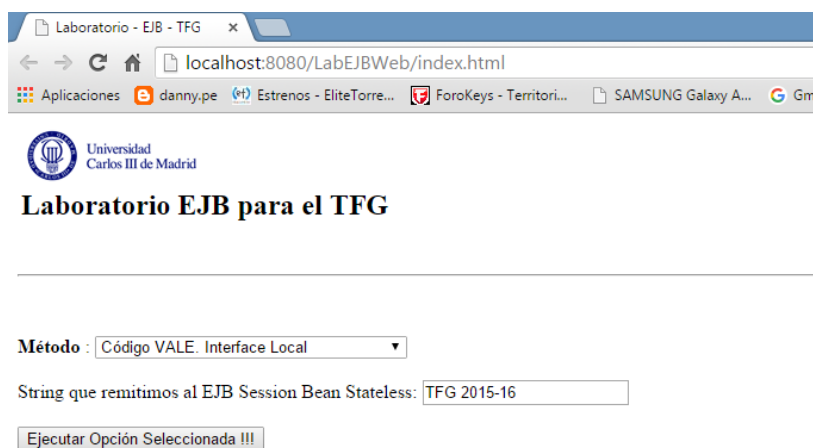


Ilustración 9 - Pedida del vale



Ilustración 10 - Obtención del vale

## MySQL



Ilustración 11 - MySQL

Su origen se remonta alrededor de la década de los noventa, cuando Michael Windenius realizaba complejas aplicaciones en lenguaje BASIC y al no encontrar un sistema de almacenamiento de archivos que le resultara satisfactorio, pensó en construir el suyo propio.

En 1995 Windenius en colaboración con David Axmark desarrollaron un producto que era el resultado de sus constantes investigaciones, con dos importantes aportaciones como son el uso del lenguaje SQL y la accesibilidad a través de Internet. Así nació MySQL y también la empresa MySQL AB.

La evolución de MySQL se produce con las sugerencias de los usuarios, canalizadas por la empresa MySQL AB, que contrata a programadores de todo el mundo para que, a través de internet, vayan ampliando y mejorando el producto. Las versiones actualizadas, se cuelgan en internet para que los usuarios puedan disponer de ellas.

En octubre de 2005, Oracle Corporation adquirió Innobase OY, que desarrolló InnoDB el motor de almacenamiento que permite a MySQL proporcionar funcionalidades como las transacciones y las claves foráneas. En 2006, MySQL emite un comunicado de prensa confirmando un acuerdo con Innobase OY de varios años. En enero de 2008, Sun Microsystems compra MySQL por 1 millón de dólares. En abril de 2009, Oracle Corporation llegó a un acuerdo para comprar Sun Microsystems [17].

MySQL tiene bastantes ventajas por ejemplo, soporta gran variedad de Sistemas Operativos, su instalación y configuración es sencilla, dispone de conectividad y seguridad, tiene velocidad para realizar las operaciones y es de bajo coste porque se puede ejecutar en una máquina con escasos recursos sin ningún problema. Además tiene una baja probabilidad de corromper datos, por el contrario no es intuitivo y un gran porcentaje de utilidades no están documentadas [18].

A continuación, se puede observar un laboratorio el cual consiste en crear una tabla e insertarle valores.

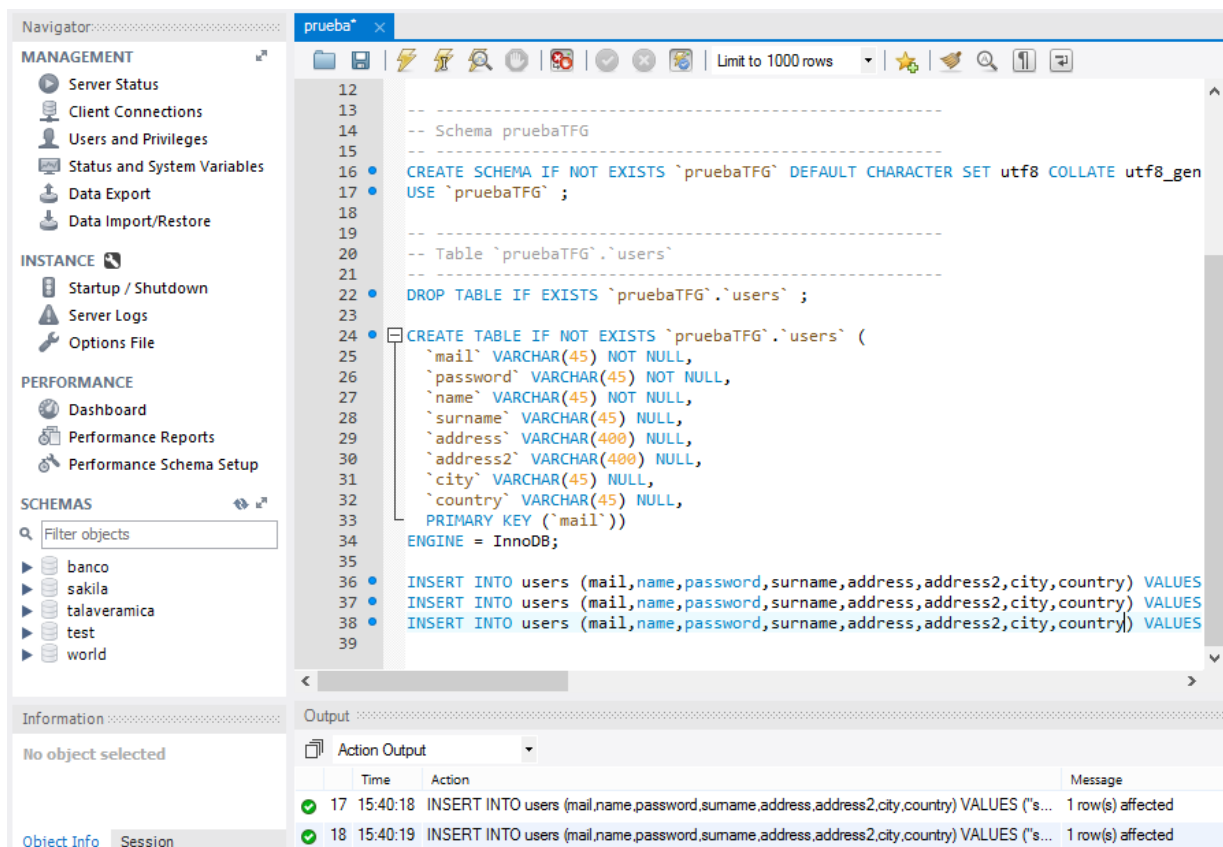


Ilustración 12 - Creación de una base de datos en MySQL

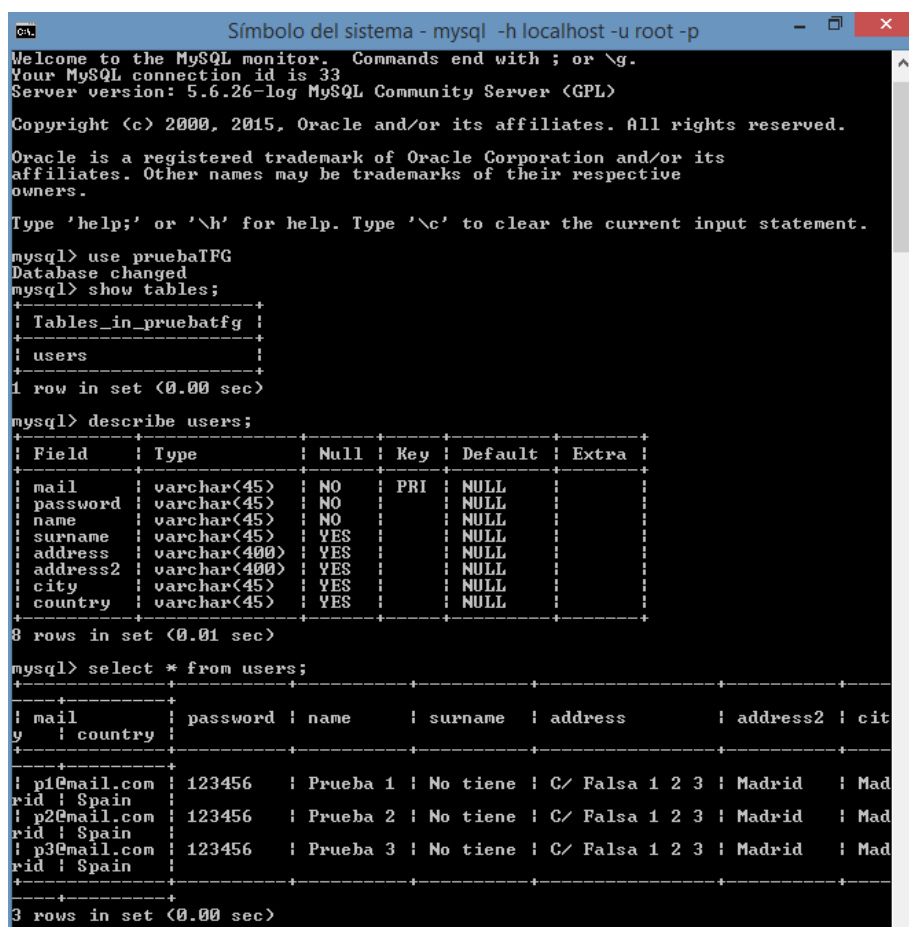


Ilustración 13 - Verificación por medio de la consola de comandos

## Transformación Digital

Cuando se habla de transformación digital hablamos de oportunidades de negocio, sabemos que es un proceso complejo, con implicaciones en la estructura de la compañía, en la cultura corporativa y en la forma de relación con el cliente. La transformación digital en el negocio eCommerce pasa por adaptarse al cliente y ser capaces de crear nuevos modelos de negocio a partir de tecnologías como el *Cloud Computing*, el *Big Data* y la movilidad.

### Cloud Computing



Ilustración 14 - Cloud Computing

La computación en la nube, conocida también como *cloud computing*, es un paradigma de computación informática que permite ofrecer servicios a través de Internet. Una de las grandes ventajas que ofrece este tipo de arquitectura computacional es que los usuarios pueden acceder a los servicios disponibles "en la nube"

de forma totalmente transparente, es decir, sin conocimientos de los recursos que usan. La información se almacena de manera permanente en servidores y se envía a cachés temporales en los equipos de los clientes (ordenadores personales, portátiles, smartphones, etc.).

El *cloud computing* no es más que un conjunto de servidores conectados a Internet encargados de atender peticiones con una alta disponibilidad. Se puede tener acceso a su información o servicio, mediante una conexión desde cualquier dispositivo móvil o fijo ubicado en cualquier lugar siempre y cuando ambos equipos (cliente y servidor) compartan la misma red. Este tipo de arquitectura distribuida permite reducir los costes y garantizar un mejor tiempo de actividad de acuerdo al contrato de garantía o nivel de servicio, comúnmente conocido como SLA (*Service Level Agreement*) adscrito entre el proveedor del servicio y el cliente.

Sus principales características pasan por una infraestructura tecnológica dinámica que se caracteriza, entre otros factores, por un alto grado de automatización, una rápida movilización de los recursos, una elevada capacidad de adaptación para atender a una demanda variable, así como virtualización avanzada y un precio flexible en función del consumo realizado, evitando además el uso fraudulento de software y la piratería.

Dependiendo del tipo de servicio que ofrece podemos encontrar los siguientes tipos de cloud:

- ~ SaaS o Software como Servicio. Ejemplos: Gmail, Google Docs, etc.
- ~ PaaS o Plataforma como Servicio. Ejemplo: Google App Engine, que permite desarrollar aplicaciones en Java o Python desplegándolas en la infraestructura que provee Google.
- ~ IaaS o Infraestructura como Servicio. Ejemplo: Amazon Web Service (AWS) que nos permite manejar máquinas virtuales en la nube, además de usarlas como almacenamiento masivo.

En resumen, un sistema cloud simplemente describe y representa a un conjunto de sistemas que están conectados a una red y proporcionan una infraestructura escalable para que los sistemas de software y los datos puedan hacer uso de ella [19].

## Big Data



Ilustración 15 - Big Data

Durante los últimos años y debido a los numerosos avances en los sistemas de información, ha sido posible la masificación del uso de aplicaciones web, móviles y sociales. Debido a esto, las grandes empresas se han visto saturadas de información y datos de sus propios clientes, y es aquí, en este

contexto, donde surge la necesidad de organizar, clasificar, cuantificar y por supuesto, explotar esta información para impulsar el beneficio económico de las propias organizaciones. Esta ingente cantidad de datos es lo que conocemos actualmente como Big Data. Pero, ¿por qué usar Big Data? Pues bien, Big Data es a día de hoy una de las tecnologías que todas las empresas quieren aplicar, tiene beneficios muy variados como la detección de fallos, conocer a los consumidores y determinar cuáles son sus tendencias incluso antes de que los propios consumidores lo sepan.

El concepto de Big Data se aplica a toda aquella información que no puede ser procesada o analizada utilizando procesos o herramientas tradicionales. El límite superior de procesamiento ha ido aumentando a lo largo de los años, en 2008 rondaban en el orden de petabytes/exabytes de datos, mientras que a día de hoy (2016) este límite computacional ronda los zettabytes.



Sin embargo, Big Data no está limitado a alguna cantidad de datos, sino que cómo apuntábamos anteriormente tiene mayor relación con la diversidad de los datos, las formas en las que pueden ser representados, sus características computacionales y la manera en que estos se relacionan/almacenan. Además de la manejabilidad de los datos, otro factor a tener en cuenta cuando hablamos de Big Data es la velocidad de análisis, acceso, búsqueda y devolución de un elemento cualquiera, pues muchas de las aplicaciones que implementan esta tecnología requieren de una respuesta lo demasiado rápida para lograr obtener la información correcta en el momento preciso.

También es importante entender que las bases de datos convencionales son una parte importante y relevante para una solución analítica, luego el uso de Big Data no hace que el resto de tecnologías queden obsoletas, al contrario, un avance en las tecnologías convencionales supone también un impulso de Big Data y viceversa [20].

### Movilidad

La movilidad es una pieza clave en la transformación digital. Apostar por esta solución implica que los clientes puedan acceder de manera rápida y ágil a su información aprovechando al máximo las funcionalidades del cloud y sacando el máximo rendimiento a los sistemas tecnológicos.

Cuando un cliente utiliza un dispositivo (smartphone, tablet, portátil, etc.) conectado a internet y realiza determinadas acciones con una aplicación permite a ese negocio recibir cierta información. Que después tratará y analizará para poder potenciar más su competencia.

### Puppet



Puppet nos da la posibilidad de automatizar tareas tediosas y desplegar rápidamente nuevos sistemas para contar con aplicaciones estables y robustas en pocos minutos. Logra liberar tiempo para trabajar en otros proyectos, garantizando la coherencia, fiabilidad y escalabilidad, y permitiendo una colaboración más estrecha entre los administradores de sistemas y los desarrolladores.

Su arquitectura Cliente/Servidor, permite centralizar y orquestar todos los despliegues y configuraciones, bajo petición de los clientes o servidores que son gestionados.

Se puede encontrar de dos formas:

- ~ De código abierto, ya que está diseñado para ayudar a los administradores de sistemas a automatizar todas esas tareas que se repiten con regularidad.
- ~ De manera comercial, incluye mejoras que no se ofrecen de manera abierta [21].

## Docker



Ilustración 17 - Docker

Su idea principal es crear contenedores ligeros y portables, convirtiéndose en una herramienta que ayuda tanto a desarrolladores, como a administradores de sistemas a probar aplicaciones software en un entorno seguro e igual al de producción, reduciendo así tiempos de pruebas y adaptaciones a cambios de hardware desde el entorno de prueba al de producción.

Docker es un proyecto de código abierto con el que fácilmente se pueden crear contenedores y se podrían llegar a definir como máquinas virtuales ligeras. Docker hace uso de las funciones de aislamiento de recursos del *kernel* de Linux para poder dar lugar a contenedores independientes, dentro de los cuales se ejecutará una única aplicación con sus respectivas dependencias, pero funcionando siempre con un único *kernel*, el de la máquina real, en lugar de virtualizar uno por cada contenedor o máquina virtual [22].

Las características principales de Docker son:

- ~ Portabilidad, todos los contenedores son portables, por lo que podemos llevarlos fácilmente a cualquier otro equipo con Docker sin tener que volver a configurar nada.
- ~ Ligereza, al no virtualizar un sistema completo, sino solo lo necesario, el consumo de recursos es mínimo. El ahorro de recursos es de alrededor de un 80%.
- ~ Autosuficiencia, Docker se encarga de todo, por lo que los contenedores tan solo deben tener lo necesario para que la aplicación funcione, por ejemplo, aquellas librerías, archivos y configuraciones necesarias para poder realizar su función.

El notable éxito de esta tecnología se puede comprobar a través de todas sus virtudes, incluida la aceptación e implicación de grandes empresas como Google o DigitalOcean que acogen este tipo de servicios en sus entornos de producción [23].

## CoreOS



Ilustración 18 - CoreOS

Es un sistema operativo ligero de código abierto basado en *cloud*. Se diseñó para proporcionar la infraestructura para los despliegues en clústeres. En lugar de instalar los paquetes vía *yum* o *apt*, CoreOS utiliza contenedores de Linux para administrar sus servicios a un nivel más alto de abstracción.

El código del servicio, como todas sus dependencias, se empaquetan dentro de un contenedor Docker que se puede ejecutar en una o varias máquinas CoreOS. Los contenedores se basan en un sistema operativo común logrando ser mucho más ligeros y más eficientes que los hipervisores. Además, descansan en una única instancia de Linux y se usan de manera acoplable, así las aplicaciones utilizan menos recursos del sistema.

En resumen, CoreOS admite un nivel de automatización elevado, facilidad de despliegue de las aplicaciones, seguridad, consistencia, fiabilidad y escalabilidad a nivel de aplicación y de sistema [24].

## Microservicios

Un microservicio consiste en desarrollar una aplicación software como una serie de pequeños servicios, cada uno ejecutándose de forma autónoma y comunicándose entre sí, por ejemplo, a través de peticiones HTTP a sus API. Normalmente hay un número mínimo de servicios que gestionan cosas comunes para los demás (como el acceso a la base de datos), pero cada microservicio es pequeño y le corresponde un área de negocio de la aplicación [25].

Como cada microservicio es independiente disponen de varias ventajas:

1. Es más fácil para los desarrolladores trabajar con ellos, ya que son relativamente pequeños y fáciles de abordar.
2. Si un microservicio falla no afecta a los demás porque son totalmente independientes. Es decir, siguen el principio de responsabilidad única.
3. Facilita el despliegue ya que no se despliega la aplicación entera sino que se despliega cada uno de los microservicios. Los despliegues son más rápidos y menos arriesgados, además reduce el tiempo de salida a producción.
4. Se puede escribir en un lenguaje de programación distinto porque sólo exponen la API al resto de microservicios.

5. Se puede acceder a los microservicios desde todo tipo de dispositivos ya que están publicados vía REST o algún tipo de RPC [26].

Los microservicios tienen muchos beneficios pero también disponen de alguna desventaja:

1. Mayor complejidad
2. Transacciones distribuidas por múltiples servicios lo que hace de un manejo más complejo.
3. Complejidad operativa en el despliegue, monitoreo y gestión.

A continuación se puede observar una imagen que compara la arquitectura monolítica con los microservicios, aclarando cualquier duda que haya podido surgir durante este apartado.

## Arquitectura Monolítica vs. Microservicios

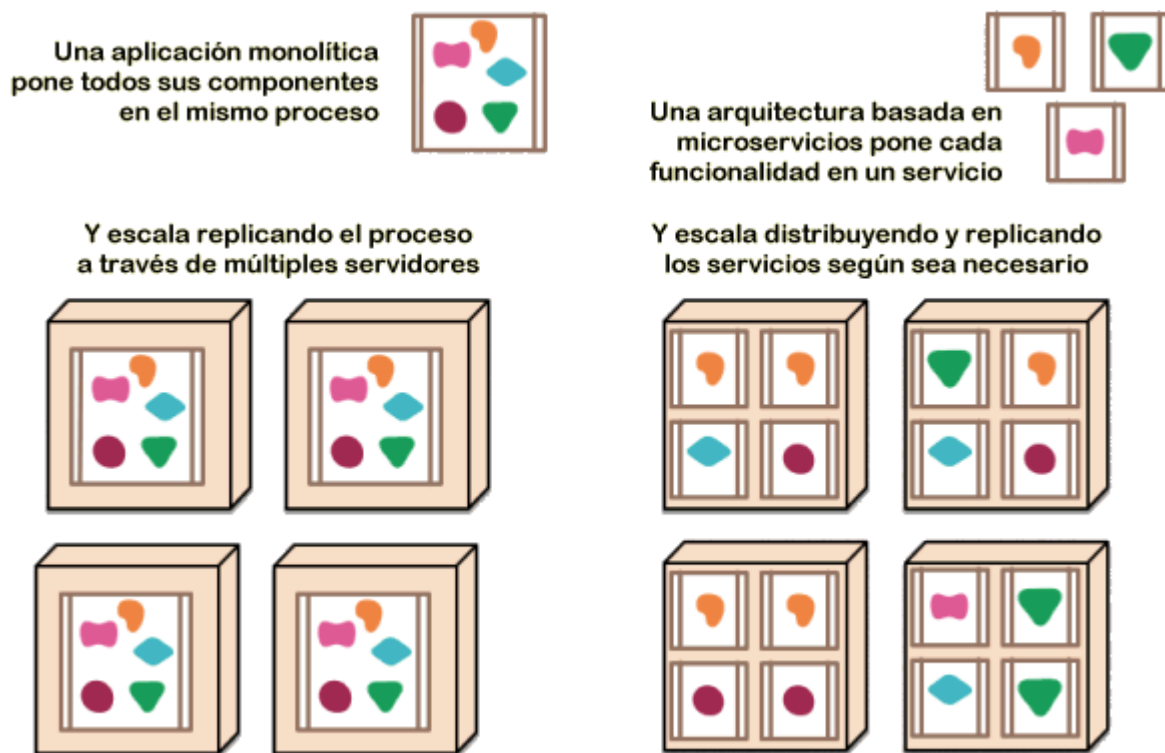


Ilustración 19 - Arquitectura Monolítica vs Microservicios

## Metodologías

Una metodología de desarrollo de software se refiere al entorno que se usa para estructurar, planificar y controlar el proceso de desarrollo de un sistema de información. Hay una gran variedad de metodologías, cada una de ellas con sus fortalezas y debilidades. Una determinada metodología no es necesariamente aplicable a todo tipo de proyectos, más bien cada tipo de proyecto tiene una metodología a la que se adapta mejor [27]. Las metodologías más usadas y extendidas se detallan a lo largo de este apartado.

### Cascada

Es el modelo más básico de todos y ha servido como bloque de construcción para los demás paradigmas de ciclo de vida. Es un proceso secuencial en el que cada fase es vista hacia abajo (como una cascada), de tal forma que el inicio de la siguiente etapa debe esperar a la finalización de la etapa previa.

La definición de este modelo fue propuesta por el Dr. Winston W. Royce en 1970 [28] pero no fue hasta los años 80 cuando el modelo se refinó, primeramente por Boehm y luego por Sommerville [29] para dar lugar a lo que hoy se conoce como modelo en cascada.

El arquetipo del ciclo de vida abarca las siguientes etapas:

- ~ El *análisis de requerimientos* consiste en recopilar las necesidades del producto y casi siempre su salida es texto.
- ~ El *diseño* se enfoca en cuatro atributos distintos del programa; la estructura de los datos, la arquitectura del software, el detalle procedimental y la caracterización de la interfaz. El proceso de diseño traduce los requisitos en una representación del software con la calidad requerida antes de que comience la codificación.
- ~ La *codificación* donde el producto de esta etapa es el código en cualquier nivel, incluido el producido por sistemas de generación automática.
- ~ La *prueba*, la cual se centra en la lógica interna del software y en las funciones externas, realizando pruebas que aseguren que la entrada definida produce los resultados que realmente se requieren.
- ~ La *verificación* es la fase donde el usuario final ejecuta el sistema, para ello el o los programadores ya realizaron exhaustivas pruebas para comprobar que el sistema no fallaba.

- ~ El *mantenimiento* es la fase en la que el software sufrirá cambios después de que se haya entregado al cliente. Estos cambios son producidos debido a los errores encontrados, a que el software se debe adaptar a cambios del entorno o a que el cliente requiera ampliaciones funcionales o de rendimiento.

La aplicación de la metodología en cascada se orienta mejor para el desarrollo de proyectos a corto plazo, de poca innovación y proyectos definitivos y detallados. Para comenzar la aplicación de la metodología en cascada, se necesita tener el análisis de los requerimientos bien definidos. El resultado del desarrollo dependerá de que estos requerimientos sean los adecuados para satisfacer la necesidad del proyecto. Se caracteriza por cumplir un orden secuencial en el desarrollo de sus tareas esto implica retardar el avance del proyecto, ya que cada etapa se inicia cuando haya finalizado la anterior, siempre y cuando se haya realizado la evaluación respectiva y resuelto los errores en caso de que los hubiera tenido. Esto implica que el proceso puede verse afectado en los costes y en su fecha de entrega, por lo que el cliente deberá tener paciencia para esperar los resultados.

## Ágil

La metodología de desarrollo ágil se refiere a los métodos basados en el desarrollo iterativo e incremental que implicará equipos multifuncionales, es decir, que no hay solamente una persona con conocimiento exclusivo para hacer algo. También implica la auto-organización, referida a que en la mayoría de los productos ágiles no hay, por ejemplo, un único jefe de proyecto responsable de asignar tareas [30].

La historia de la metodología ágil obtuvo su reconocimiento en el año 2001 cuando destacados y conocidos miembros de la ingeniería del software escribieron en Utah el Manifiesto Ágil. El objetivo consistía en establecer valores y principios rápidamente, para permitir a los equipos el desarrollo de software, poder responder a cambios que pudiesen surgir a lo largo del proyecto. Se pretendía ofrecer una alternativa a los procesos de desarrollo de software tradicionales, caracterizados por ser rígidos y dirigidos por la documentación que se genera en cada una de las etapas de desarrollo.

Uno de los principales focos de aplicación de las metodologías ágiles son los proyectos tecnológicos. Cada una de ellas tiene sus fortalezas y sus debilidades, pero no son excluyentes. En cada proyecto podemos adoptar una, o varias, en función de las

características del propio proyecto y del equipo. Entre las metodologías ágiles más usadas encontramos [31]:

- ~ *Scrum* es un marco de trabajo que nos proporciona una serie de herramientas y roles para, de una forma iterativa, poder ver el progreso y los resultados de un proyecto y así minimizar riesgos.
- ~ *Kanban* se basa en que el trabajo en curso debería limitarse y sólo deberíamos empezar con algo nuevo cuando un bloque de trabajo anterior haya sido entregado o haya pasado a otra función posterior de la cadena.
- ~ *XP (Extreme Programming)* se centra en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores y propiciando un buen clima de trabajo.
- ~ *SAFe (Scaled Agile Framework)* es un marco de trabajo, probado y público para aplicar *Lean* y *Agile* a escala empresarial, permitiendo trasladar los beneficios de estos paradigmas a proyectos de gran envergadura que son soportados por grandes equipos de trabajo. En definitiva, estructura los proyectos de software para que den un soporte ágil a los procesos de negocio, reduciendo el “Time to market”, aumentando la productividad y ayudando a desarrollar más ventajas competitivas.

En un proyecto ágil el diseño, el desarrollo y las pruebas se realizan de manera continua. De esta forma, se lleva la iteración al extremo:

1. Se busca dividir las tareas del proyecto en incrementos con una planificación mínima y de corta duración (entre 1 y 4 semanas).
2. Cada iteración suele concluir con un prototipo operativo. Al final de cada iteración, se obtiene un producto entregable que es revisado junto con el cliente, posibilitando la aparición de nuevos requisitos o perfeccionando los existentes.

Las metodologías ágiles suelen ofrecer una solución a una gran cantidad de proyectos, además una de las cualidades más destacables es su sencillez, tanto en aprendizaje como en su aplicación, reduciéndose así los costes de implantación en un equipo de desarrollo. Pero estas metodologías están dirigidas a equipos pequeños o medianos (de 3 a 20 personas), donde el entorno físico debe ser un ambiente que permita la comunicación y colaboración entre todos los miembros del equipo durante todo el tiempo. Cualquier resistencia del cliente o del equipo de desarrollo hacia las prácticas y principios pueden llevar al fracaso.

## Análisis del Negocio

El mundo eCommerce está muy solicitado hoy en día y con tiendas virtuales muy parecidas a lo que se pretende desarrollar en este proyecto. En este apartado se pretende conocer contra quien se compite, los datos que podemos saber de ellos, dónde estamos dentro de este mundo y qué podemos hacer para mejorar. A continuación, se muestran 4 negocios eCommerce que compiten con el negocio de este proyecto.

### Amazon

Es una empresa estadounidense de comercio electrónico, dedicada a la venta de productos por internet. Empezó con la venta de libros y a día de hoy se ha convertido en una de las empresas más grandes de Internet con toda clase de productos en sus listas.

En 1994, Jeff Bezos creó "Cadabra.com", él había trabajado como programador en Wall Street y su objetivo fue el de crear una tienda competitiva usando internet. En 1995, lanzó su empresa y cambió el nombre de la empresa por Amazon.

Lo que hace a Amazon especial, es su esfuerzo por personalizar el servicio que proporciona a cada usuario. Dicho de otro modo, el usuario tiene la sensación de que tiene un vendedor personal preparado para ayudarlo en cualquier momento.

Evaluando la web cabe destacar su buena estructuración, ya que puedes buscar por departamentos (Deporte, Libros, Videojuegos, Electrónica, etc.), por productos más vendidos, por ofertas y productos relacionados. Destacar que no solo ofrece información relacionada con ellos, sino que también ofrece información sobre los métodos de pago, las políticas de envíos, la gestión de la compra, etc.



Ilustración 20 - Tienda Amazon



La información que se puede obtener de un producto es muy completa, porque el usuario dispone de varias imágenes para observar con detenimiento el producto, comentarios de otros usuarios, valoraciones y la descripción del mismo. Amazon se basa en experiencias anteriores de compra para recomendar al usuario productos relacionados con sus compras. No se puede pasar por alto su capacidad de publicidad, ya que es mucho mayor que el del resto de competidoras.

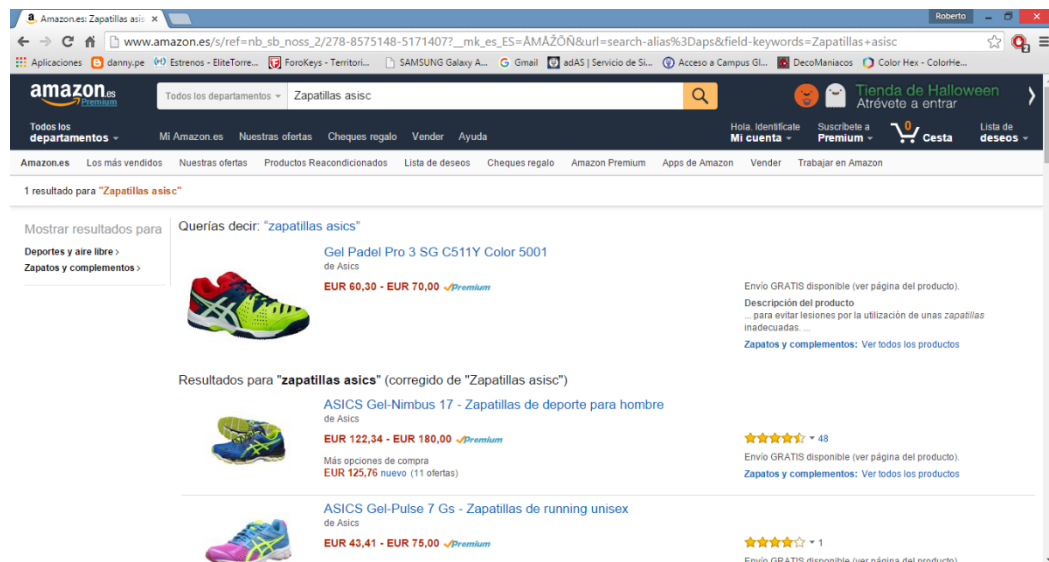


Ilustración 21 - Búsqueda de un producto en Amazon

## Zalando

Es una tienda online de calzado y moda en Europa. Está en pleno proceso de expansión pero que ya se sitúa dentro del top 10 de negocios eCommerce. Su objetivo es que el usuario disfrute de un excelente Servicio de Atención al Cliente y de un amplio surtido de productos.

La empresa con sede en Berlín fue fundada en 2008 por Robert Gentz y David Scheneider. Tras el rotundo éxito cosechado en Alemania, no dudaron en expandirse a otros países europeos como Austria, Francia, Italia, España, Bélgica, etcétera.

Zalando lleva alrededor de 2 años en el mercado español, y en este tiempo, ha conseguido hacerse un hueco en el corazón y el bolsillo de todos los clientes españoles, gracias a su enorme oferta de productos, condiciones de compra, estructura y diseño.

Las condiciones de compra son las mejores, ya que el cliente obtiene un buen número de cláusulas ventajosas:

- ~ Envío gratuito.
- ~ Envío a domicilio.

- ~ Devoluciones gratuitas.
- ~ Pago a través de tarjeta de crédito, débito, transferencia, PayPal y contra reembolso.
- ~ Atención telefónica (gratuita), a través de email y/o redes sociales.

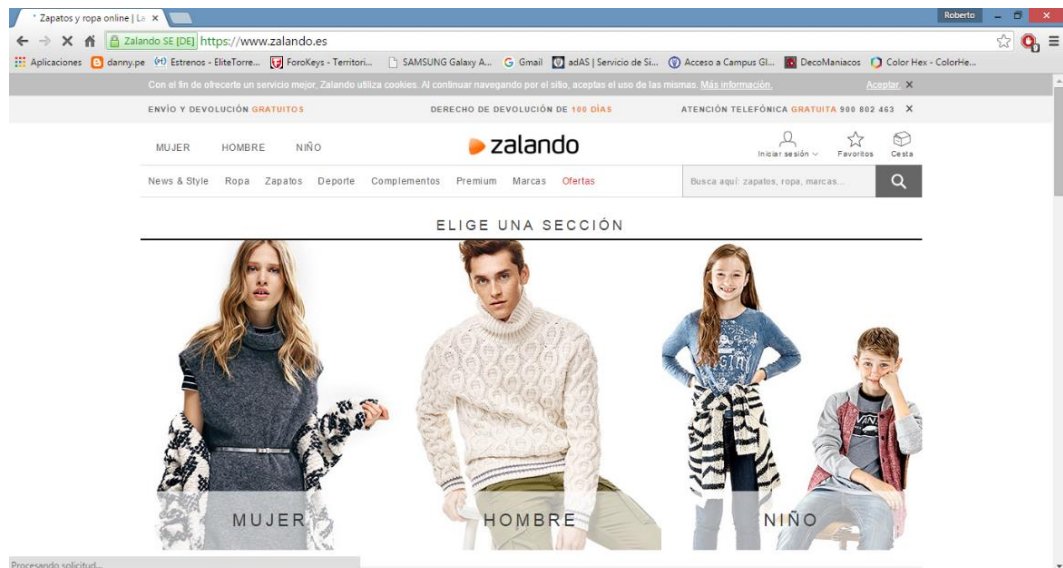


Ilustración 22 - Tienda Zalando

Cabe destacar que es una web con una distribución visual muy clara y fácil de usar, los menús son muy prácticos y encontrar cualquier producto es bastante sencillo porque dispone de amplísimo catálogo. Aunque no todo es perfecto, su principal punto débil es la distribución de los productos entre 5 y 7 días.

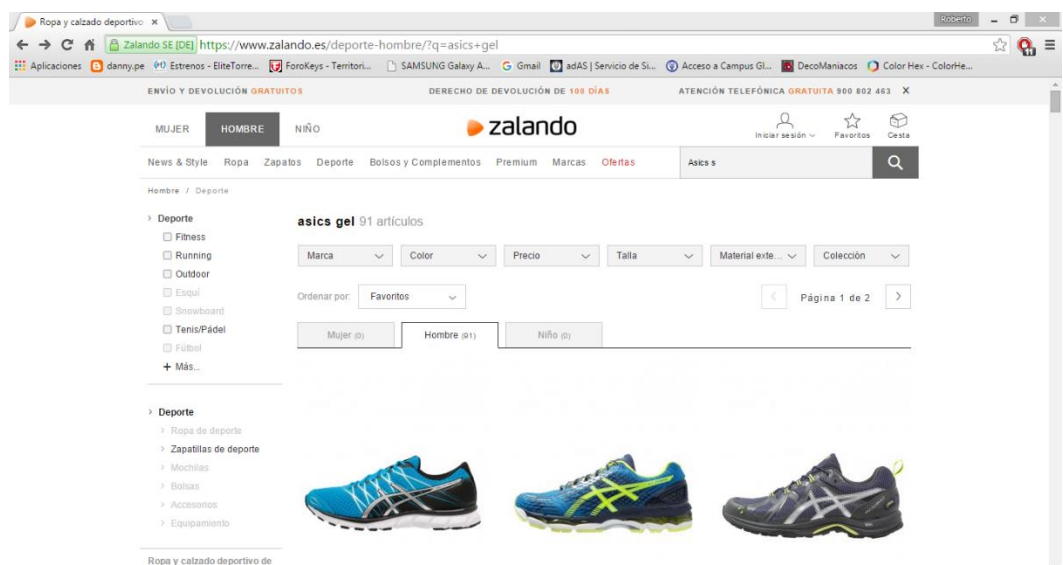


Ilustración 23 - Búsqueda de un producto en Zalando

## El Corte Inglés

Es el principal grupo de distribución comercial en España, y se sitúa entre los mejores en el ranking mundial por volumen de ventas. El corte inglés es además una de las marcas más prestigiosas y valoradas de España y del mundo.

El Corte Inglés se posiciona como el portal online líder en comercio electrónico en España, por encima de grande sites norteamericanas.

¿Por qué el Corte Inglés es líder en el negocio eCommerce en España? La respuesta es muy sencilla, porque han logrado convertir lo físico en virtual. Es decir, han cumplido diversos objetivos:

- ~ Una atención al cliente de calidad, como la que se realiza en tienda.
- ~ Un gran catálogo de productos.
- ~ Comprar un producto y poder devolverlo en una tienda física.
- ~ Cualquier duda que surja sobre la web se puede hacer online o en la oficina de atención al cliente de cualquier superficie del Corte Inglés.



Ilustración 24 - Tienda del Corte Inglés

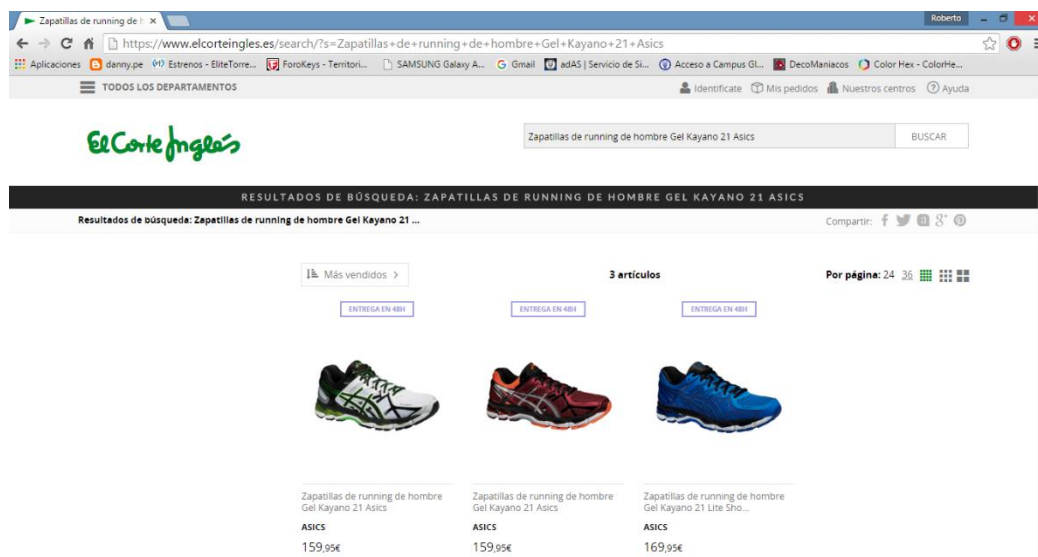


Ilustración 25 - Búsqueda de un producto en la tienda del Corte Inglés

## Under Armour

La empresa con sede está en Baltimore, Maryland, fue fundada en 1996 por Kevin Plank, un antiguo capitán del equipo americano de la universidad de Maryland. Todo empezó cuando este jugador de fútbol americano vio la posibilidad de crear camisetas transpirables con las que poder facilitar el desarrollo de cualquier actividad física. Fue así como en muy poco tiempo, los ingresos de su compañía subieron estrepitosamente y tuvieron que aumentar la producción de estas camisetas.

Under Armour es una marca fundamentalmente deportiva, con prendas y calzado de gran rendimiento. Pese a su juventud, esta compañía ha sabido ganarse en muy poco tiempo la confianza de numerosos adictos al deporte, que han decidido incluir esta marca en su indumentaria deportiva.

Su portal online destaca por la buena estructuración de los productos, estos pueden ser encontrados en categorías como accesorios, oferta, prendas o calzados deportivos. Destacar que en el menú se puede buscar por sexo y encontrar productos relacionados con él.

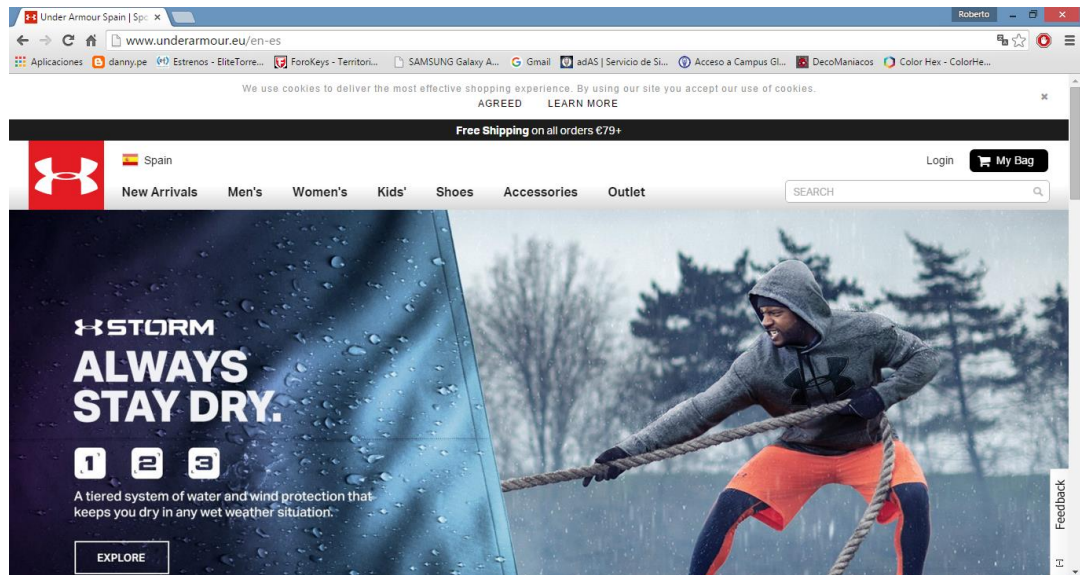


Ilustración 26 - Tienda Under Armour

La información que ofrece al cliente del producto es bastante completa, porque además de poder elegir su color y talla, dispone de una descripción detallada, de comentarios y valoraciones de los demás usuarios. El punto débil de esta web es la atención al cliente.

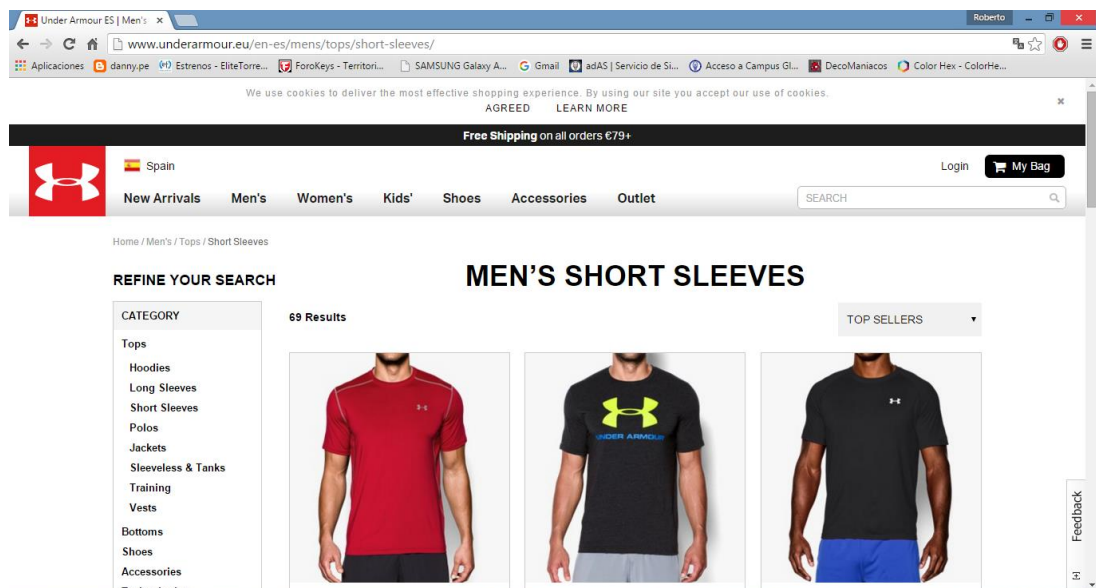


Ilustración 27 - Búsqueda de un producto Under Armour

## Capítulo III – Buscando una Solución Software

Tras haber analizado las tecnologías referentes, pasamos a exponer brevemente cual es la idea del cliente. Como en cualquier proyecto software el cliente por lo general no conoce o no tiene una idea específica de cómo va a funcionar el sistema, así que se procede a realizar una entrevista en la que se negocie los diferentes puntos que afectan al mismo. La primera reunión con el cliente tuvo lugar el día 12 de Noviembre de 2015 y su finalidad fue conocer más en detalle lo que requería. El contenido de esta reunión queda expuesto en los sucesivos apartados.

### Idea Original

Tras las presentaciones oportunas, el cliente nos cuenta lo que necesita. Su idea queda recogida en las siguientes líneas:

*“Una aplicación web que facilite la operativa de un site/tienda de venta de ropa, calzado y accesorios de deporte. La tienda final de venta contendrá un catálogo con la ropa, calzado y accesorios deportivos, puestos a disposición por un conjunto de proveedores. También necesita que el sistema para el cobro se integre con distintas pasarelas de pago y disponga de una gestión financiera que incluya la conciliación de las ventas. Hace hincapié en que se envíen mensajes desde la aplicación, pero de dos maneras diferentes, primero un mensaje a los proveedores cuando el producto es denegado reportándole por que se deniega y segundo, un mensaje a los usuarios el cual contenga un pdf con toda la información respectiva a la compra que acaba de realizar. La plataforma de desarrollo para la aplicación ha de ser Java”*

Si analizamos las peticiones del cliente parece razonable pensar que se pueden cumplir todas las restricciones sin excepción. Es el momento de realizar un proceso de síntesis para conocer y detallar mejor las restricciones del sistema.

### Restricciones Contractuales y Problemas Derivados

Las restricciones que nos impone el cliente son, en definitiva, las siguientes:

- ~ **El proveedor solo podrá ver sus productos y ajustar los precios** → Por tanto, el proveedor solo podrá añadir sus productos, modificarlos y ajustarse a los precios que imponga el administrador. Es verdad que el proveedor puede poner el precio que crea conveniente para un artículo pero será el administrador quien lo apruebe. Si es denegado recibirá un correo en el que se le notificará porque ha sido denegado.



- ~ **No es necesaria la implementación de un servidor** → Al no ser necesaria la implementación de un servidor, podemos centrarnos en otros aspectos importantes del proyecto.
- ~ **Los usuarios deben registrarse antes de realizar una compra** → Cuando un usuario de la aplicación se decide por comprar un artículo de la tienda, tiene que acceder al sistema, la única manera es registrándose (si aún no lo está) o accediendo a ella mediante su usuario y contraseña.
- ~ **La plataforma de desarrollo para la aplicación web ha de ser Java** → No supone ningún problema a la hora de trabajar en el desarrollo de la aplicación.

## Selección de la Solución Software

Llegados a este punto el cliente nos da total libertad a la hora de diseñar su solución software, por ello, tenemos que analizar las tecnologías disponibles expuestas en el capítulo anterior en busca de aquellas que nos permitan implementar una solución sencilla, simple, eficaz y eficiente.

### Servidor de Aplicaciones

A petición del cliente el servidor no será implementado, es decir, no se creará un servidor de cero debido a que perderíamos demasiado tiempo en su construcción. Lo que se va a hacer es evaluar y analizar los diferentes servidores en busca del mejor para nuestro proyecto. Para cerciorarnos de que seleccionamos la opción correcta iremos descartando los servidores no viables en función de sus características.

En primer lugar se descarta el servidor JBoss por problemas de rendimiento y saturación en el uso de EJB en el lado del cliente y el servidor. También una razón de peso por lo que es descartado es porque usa servicios globales innecesarios que reducen el rendimiento, la memoria y el espacio en disco.

En segundo lugar queda descartado el servidor Jetty, aunque puede ser más fácil de integrar que cualquier otro servidor su gran problema es que se centra en ser un buen contenedor web, y lo que se quiere en esta aplicación es un servidor web completo y eficiente.

El último servidor que es descartado es Tomcat, aunque es bastante ligero y su complejidad no es tan elevada como las de otros servidores. El principal problema de Tomcat es que es sólo un contenedor web, es decir, que no utiliza todas las Java EE que necesitamos en nuestra

aplicación. Otro problema que se le puede achacar es que suele fallar cuando la aplicación es muy compleja.

Así pues, queda seleccionado como servidor web Glassfish debido a su estabilidad, a que es un contenedor Java EE y a que maneja la ejecución de EJB, JMS y JTA. Además, es la implementación de referencia para J2EE y está desarrollado por Sun Microsystems.

## Arquitectura

La selección de una arquitectura adecuada va a sentar las bases del desarrollo de un proyecto. En este caso, no se dan alternativas de arquitecturas diferentes, ya que la arquitectura C4 (que se desarrolla más adelante) quedó impuesta por el tutor del proyecto en busca de afianzar, ampliar y profundizar en los conocimientos y competencias que se han ido adquiriendo durante la etapa universitaria. Además, potenciar el desarrollo del proyecto haciendo uso de las tecnologías que se van a ver a continuación:

- ~ **JAVA EE:** plataforma de programación Java para desarrollar y ejecutar la aplicación alojada en el servidor. En combinación con una base de datos MySQL se puede construir un sistema rápido y eficaz.
- ~ **Java Persistence API:** más conocida por sus siglas JPA, es la API de persistencia desarrollada para la plataforma Java EE. Fue elegida como modelo de persistencia ya que está basado en POJO's y mapea bases de datos relacionales.
- ~ **MySQL:** al disponer de alojamiento gratuito en el servidor web y soportar este tipo de base de datos, se ha elegido como base de datos.
- ~ **JSON:** se ha elegido por la simplicidad en el intercambio de información que se produce entre cliente y servidor.
- ~ **REST:** arquitectura para la comunicación entre el cliente y el servidor. Dado que REST trabaja con el protocolo HTTP, la simplicidad que posee en comparación con otros métodos de comunicación como CORBA, SOAP o WDSL hace que su uso sea muy utilizado en los sistemas de información de hoy en día.



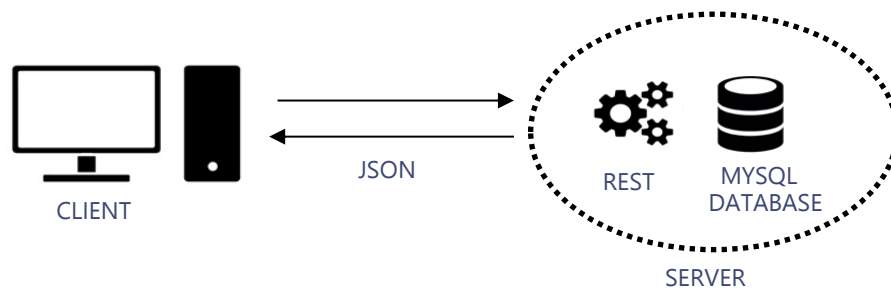


Ilustración 28 – Funcionamiento de las tecnologías en la Aplicación Web

La arquitectura se va a basar en la técnica de modelado de diagramas c4 (o las cuatro C: Contexto, Contenedores, Componentes y Clases), ya que todo sistema debe funcionar dentro de un contexto y tiene que interactuar con otros sistemas de tecnología informática. Este sistema puede ser modelado como un conjunto de contenedores, o entornos de ejecución, en los cuales se ejecutan componentes de software. A su vez, los componentes de software podrían implementarse como una agregación de elementos, es decir, como clases o interfaces. A continuación, se procede los pasos de la arquitectura:

1. **Diagramas de Contexto:** sirve para representar los límites del sistema, es decir, permite distinguir lo que es el sistema y su entorno. Ayuda a definir lo que hace y lo que no hace parte del sistema. Para elaborarlo, colocamos en el centro el sistema a construir y lo rodeamos de los tres tipos de entidades que nos podemos encontrar:
  - a. Los demás sistemas con los que interactúa.
  - b. Los usuarios del sistema.
  - c. Los centros de información que necesita.

El foco debe estar en los usuarios y sistemas de software en lugar de tecnologías, protocolos y otros detalles de bajo nivel.

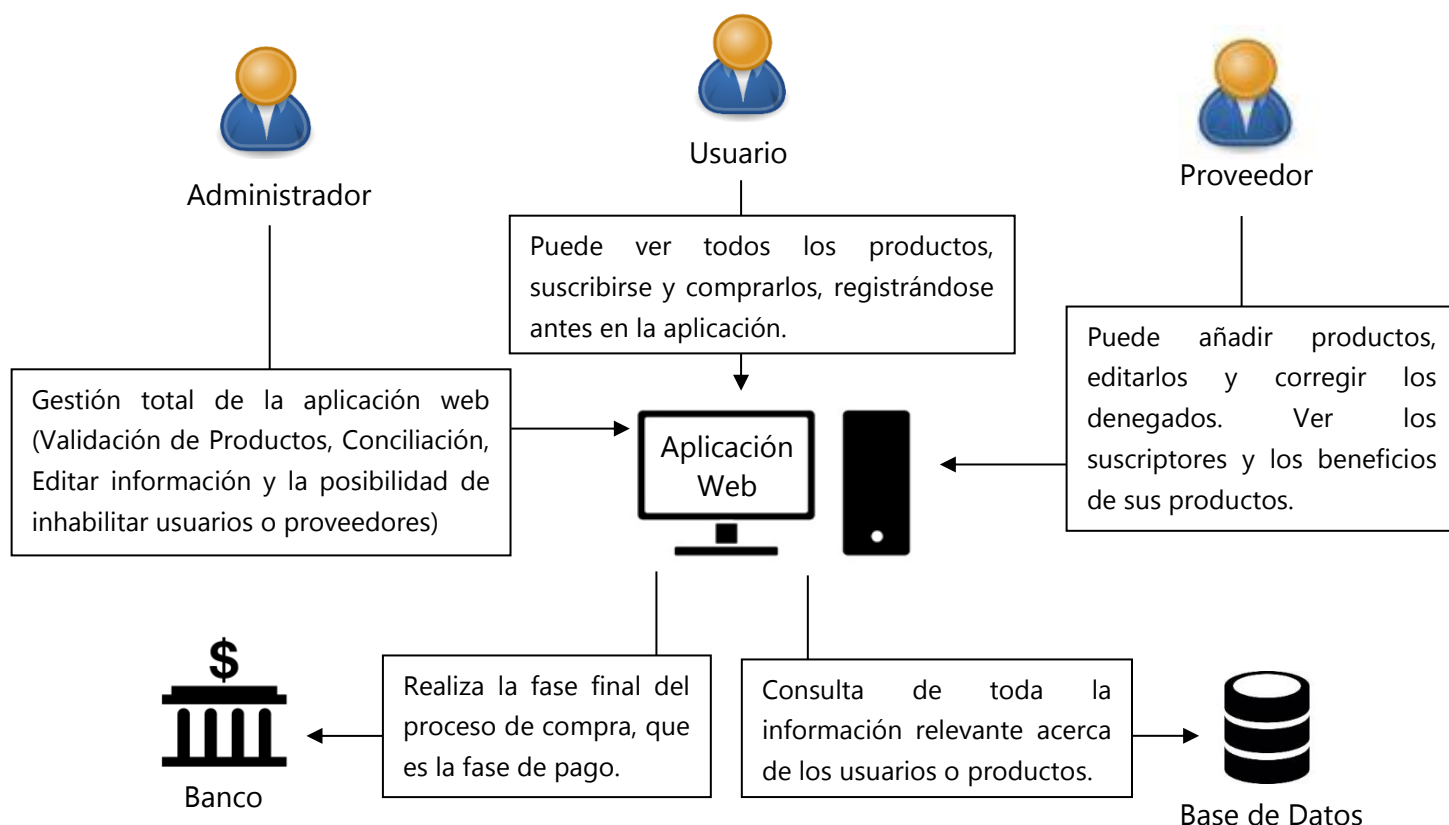


Ilustración 29 - Diagrama de Contexto

2. **Diagramas de Contenedores:** una vez entendido cómo el sistema general encaja en el diagrama de contexto, puede ser muy útil ilustrar las opciones de tecnología de alto nivel con un diagrama de contenedores. Esencialmente, un contenedor es cualquier elemento que puede albergar datos o código. Este diagrama muestra en alto nivel la arquitectura de software y cómo se distribuyen las responsabilidades a través de él. Es un diagrama de tecnología de alto nivel enfocada para desarrolladores software.

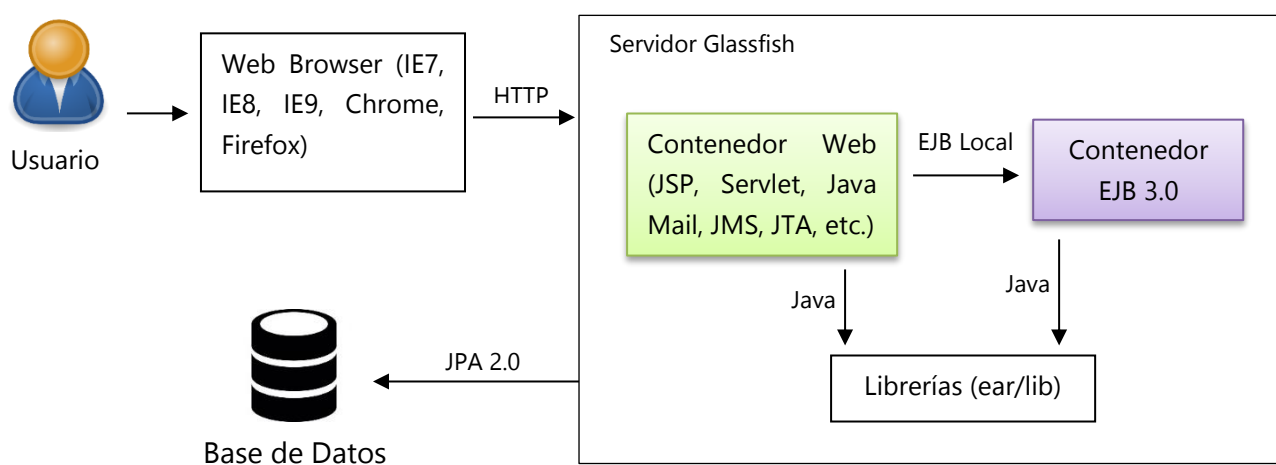


Ilustración 30 - Diagrama Contenedor Usuario

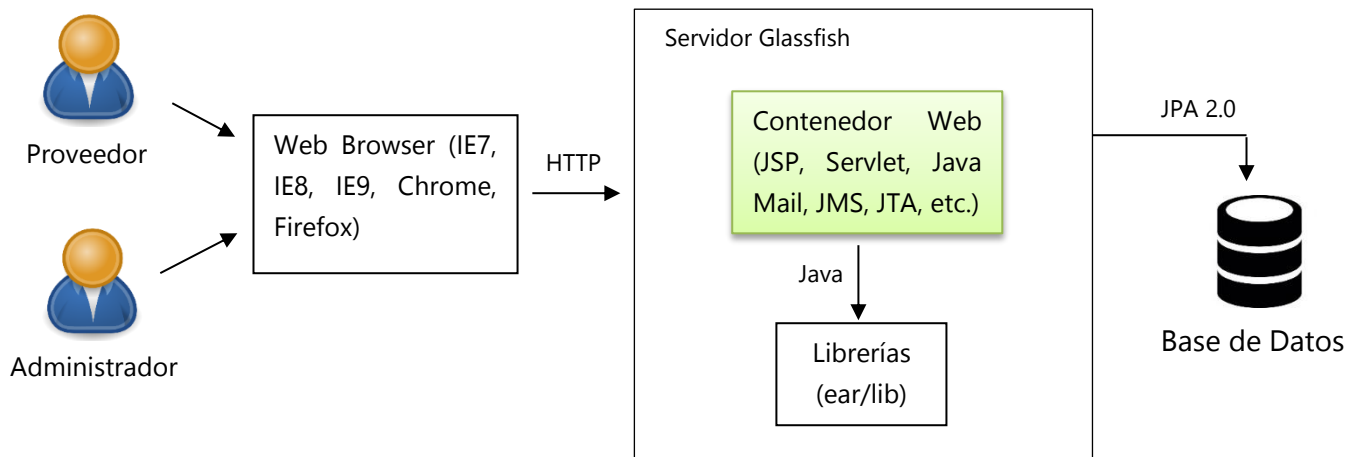


Ilustración 31 - Diagrama Contenedor Proveedor/Administrador

3. **Diagrama de Componentes:** trata de descomponer aún más todo lo que se ha podido observar en el diagrama de contenedores, es decir, cada contenedor se va a descomponer en los principales componentes lógicos y sus interacciones. En vez de mostrar un diagrama el cual puede inducir a errores, en el *Capítulo IV - Diseño del Sistema* se va a poder encontrar un apartado denominado "estructuración del código" que cuenta con detalle los componentes de los contenedores y sus relaciones.
4. **Diagrama de clases:** este tipo de diagramas es opcional, pero a veces dibujar uno o más diagramas de clase UML ayudan a ilustrar los detalles de implementación de un componente específico.

## Producto Final

Tras analizar las posibilidades de implementación y haber decidido las características del sistema, el producto final contará con los siguientes componentes:

- ~ Servidor, ejecutable desde cualquier PC si es para realizar una demostración de la aplicación web o si es utilizado a nivel comercial.
- ~ Aplicación Web en Java que permite a los usuarios ver, informarse y comprar aquellos artículos que desean.

## Capítulo IV – Diseño del Sistema

A lo largo de este capítulo se especifica el diseño que se ha realizado para el sistema cliente y el sistema servidor.

### Cliente

Por parte del sistema cliente, se expondrá el diseño que se ha realizado para estructurar el código y se mostrará el prototipo de la aplicación Web.

### Estructura del Código

Como se comentó anteriormente, ahora vamos a explicar con detalle los componentes de los contenedores (*Diagrama de Componentes*). Su estructuración ha permitido una mejor lectura del código e identificación de las clases, tanto para el presente de la aplicación como para futuros avances del proyecto. Se ha estructurado de la siguiente manera:

- ~ *Cuatro Enterprise Application Project* (Usuario, Proveedor, Administrador y Banco).
- ~ *Dos JPA Project* (GoalToRun y Banco)
- ~ *Un EJB Project* (Vales)

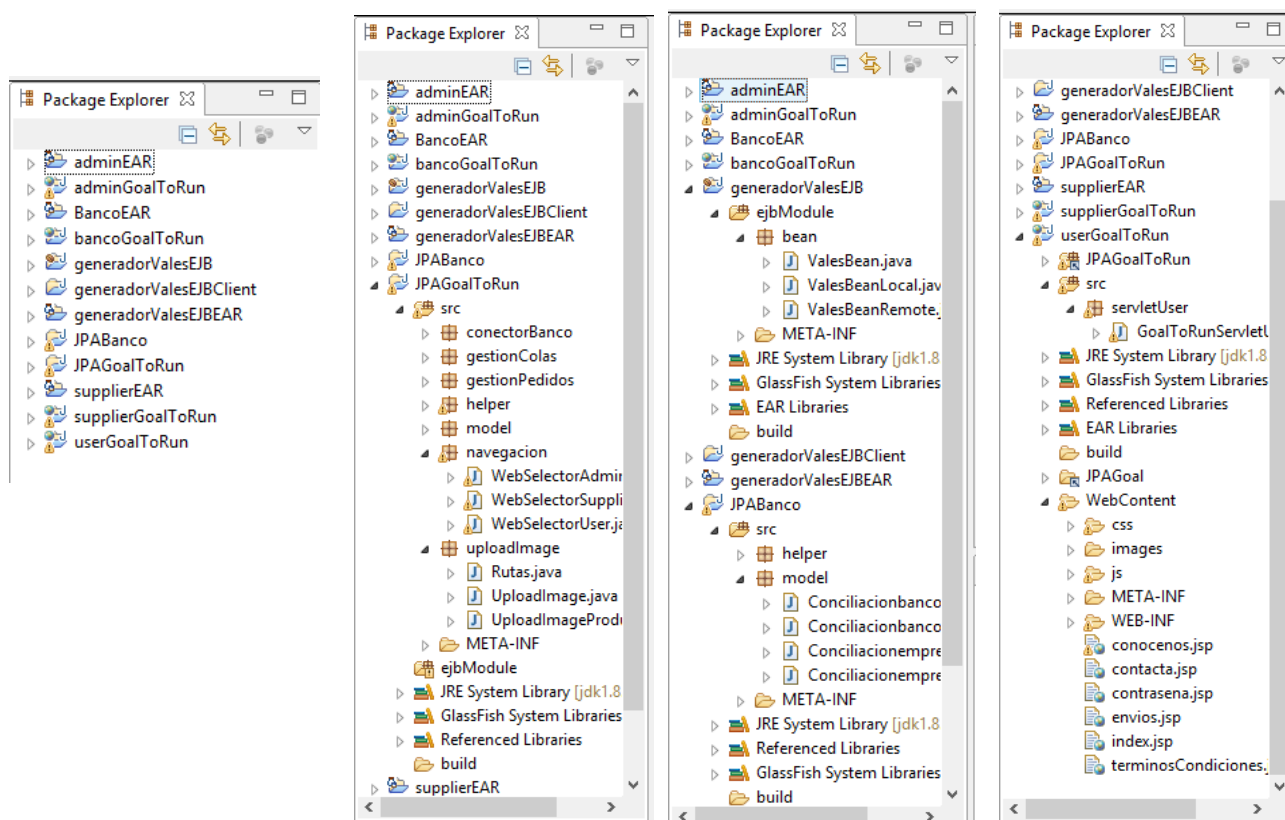


Ilustración 32 - Estructuración del Código

Una vez que conocemos como se estructura la aplicación web, se va a detallar el objetivo de cada uno de los proyectos expuestos anteriormente y se explicarán cada uno de los paquetes contenidos en los dos *JPA* porque es ahí donde radica el funcionamiento completo de la aplicación web.

- ~ **Usuario:** proyecto encargado de la creación del portal web del usuario, por medio de la creación, gestión y redirección de las jsp a su servlet.
- ~ **Proveedor:** proyecto encargado de la creación del portal web del proveedor, por medio de la creación, gestión y redirección de las jsp a su servlet.
- ~ **Administrador:** proyecto encargado de la creación del portal web del administrador, por medio de la creación, gestión y redirección de las jsp a su servlet.
- ~ **Banco:** proyecto encargado de la creación de la página web del banco, por medio de la creación, gestión y redirección de las jsp a su servlet. Además, cuenta con 4 paquetes, que se detallan a continuación:
  - **CodigoOperacion:** clase que genera un código de operación, es decir, por cada pedido realizado y pagado al banco éste le da un código que lo identifica.
  - **Conciliación:** almacena las clases que interactúan con la base de datos del banco sobre la conciliación de la empresa y proveedor.
  - **Pagos:** este paquete almacena las clases que realizan el pago al banco.
  - **TarjetaCredito:** clase que comprueba que el número de la tarjeta de crédito sea correcto.
- ~ **JPAGoalToRun:** proyecto encargado de todas las gestiones que se producen en la aplicación web, cuenta con una serie de paquetes que se detallan a continuación:
  - **ConectorBanco:** almacena dos clases, la primera permite que el pedido sea aceptado y la segunda realiza la conciliación de la empresa y proveedor.
  - **GestionColas:** almacena las clases que gestionan las colas para tratar los pedidos de una manera síncrona.
  - **GestionPedidos:** este paquete contiene la clase que gestionan el pedido, es decir, aquella que lo crea, que genera el vale descuento y que se conecta con el banco.
  - **Helper:** contiene clases encargadas de realizar consultas en la base de datos, de crear los emails que se envían desde la aplicación y por lo tanto incluye la

creación del pdf del pedido. También incluye una clase importante que es aquella que cifra datos sensibles como, por ejemplo, la contraseña.

- **Model:** contiene todas las clases que gestionan los componentes que representan la información.
  - **Navegacion:** contiene las clases más importantes de la aplicación web, en ellas, se realizan todas las funcionalidades que tienen relación con el proyecto usuario, proveedor y administrador.
  - **UploadImagen:** este paquete contiene las clases encargadas de cargar las imágenes del ordenador y almacenarlas en la base de datos, también realiza la lectura de la imagen de la base de datos y la muestra en la interfaz de la aplicación.
- ~ **JPABanco:** proyecto encargado de todas las gestiones que tienen que ver con el banco, cuenta con una serie de paquetes que se detallan a continuación:
- **Helper:** este paquete contiene todas las clases encargadas de interactuar con la base de datos del banco.
  - **Model:** contiene todas las clases que gestionan los componentes que representan la información.
- ~ **GeneradorValesEJB:** este proyecto es el encargado de realizar la gestión de los vales. El contenido de su paquete se detalla a continuación:
- **Bean:** almacena las clases que generan los vales descuento en función del dinero gastado en el pedido. Se genera también un vale de expiración, cuyo objetivo es “enganchar” al usuario de alguna manera para que vuelva a comprar en la página.

## Prototipo de Aplicación

A lo largo de este apartado se podrá observar el diseño de las páginas que componen la aplicación web, así como el flujo que se puede seguir entre las mismas. Como se comentó anteriormente, este negocio virtual cuenta con tres portales: el de los usuarios, el de los proveedores y el del administrador. A continuación, se describen esas páginas para cada uno de los portales.

## Usuario

El usuario de la aplicación web tendrá a su disposición diferentes páginas, las cuales son descritas a continuación:

- ~ **Buscador:** página que muestra el producto/s que el cliente está buscando.
- ~ **Carrito:** página que añade los productos a comprar, con toda la información necesaria.
- ~ **Confirmación de Compra:** página con el código del pedido, el descuento si superas "x" dinero y un vale de caducidad.
- ~ **Conocenos:** página con información sobre la empresa, para que el usuario nos conozca un poco más.
- ~ **Contacta:** página que da información sobre cómo ponerse en contacto con la empresa.
- ~ **Contraseña:** página para recuperar la contraseña.
- ~ **Crear Cuenta:** página para que el cliente pueda crearse una cuenta para comprar los artículos que desee.
- ~ **Datos Compra:** página que muestra la información del usuario, los datos de envío y los datos de pago.
- ~ **Editar Perfil:** página donde se puede editar la información del usuario.
- ~ **Envíos:** página que contiene información relativa a los pedidos.
- ~ **Histórico:** página que le muestra información relativa a los productos que ha comprado en nuestra web.
- ~ **Index:** página de inicio de la aplicación.
- ~ **Login:** página de inicio de sesión.
- ~ **Perfil:** página que muestra información relevante al usuario.
- ~ **Producto:** página que da información relativa a un producto.
- ~ **Productos:**
  - **Hombre (Ropa, Deportivas):** páginas que muestran ropa y deportivas de hombre.
  - **Mujer (Ropa, Deportivas):** páginas que muestran ropa y deportivas de mujer.
  - **Niño (Ropa, Deportivas):** páginas que muestran ropa y deportivas de niño.

- ~ **Resumen:** página de confirmación de la compra.
- ~ **Suscripciones:** página que nos muestra las suscripciones de productos.
- ~ **Términos y Condiciones:** página con información muy relevante, como por ejemplo las condiciones de uso, la privacidad, la propiedad intelectual, etc.





Historico

Conocenos

Envios



Ilustración 33 - Flujo Aplicación Web Usuario

## Proveedor

El proveedor de la aplicación web tendrá a su disposición diferentes páginas, las cuales son descritas a continuación:

- **Añadir Producto:** página para añadir un producto el cual debe ser aceptado por el administrador para que se ponga en venta.
- **Conócenos:** página con información sobre la empresa, para que el usuario nos conozca un poco más.
- **Contacta:** página que da información sobre cómo ponerse en contacto con nosotros.
- **Contraseña:** página para recuperar la contraseña.
- **Editar Perfil:** página donde se puede editar la información del proveedor.
- **Editar Producto:** página para modificar alguna característica del producto.
- **Envíos:** página que contiene información relativa a los pedidos.
- **Facturación:** página que da información al proveedor de los beneficios que obtiene de sus productos.
- **Index:** página de inicio de la aplicación.
- **Login:** página de inicio de sesión.
- **Mis Productos:** página que muestra únicamente los productos que el proveedor tiene en venta.
- **Perfil:** página que muestra información relevante al proveedor.
- **Producto:** página que da información relativa a un producto.
- **Productos Denegados:** página que muestra los productos que han sido denegado por el administrador, por el motivo que le puso en el correo electrónico.
- **Suscriptores:** página que muestra los suscriptores que tiene el proveedor de sus productos.
- **Términos y Condiciones:** página con información muy relevante, como por ejemplo las condiciones de uso, la privacidad, la propiedad intelectual, etc.



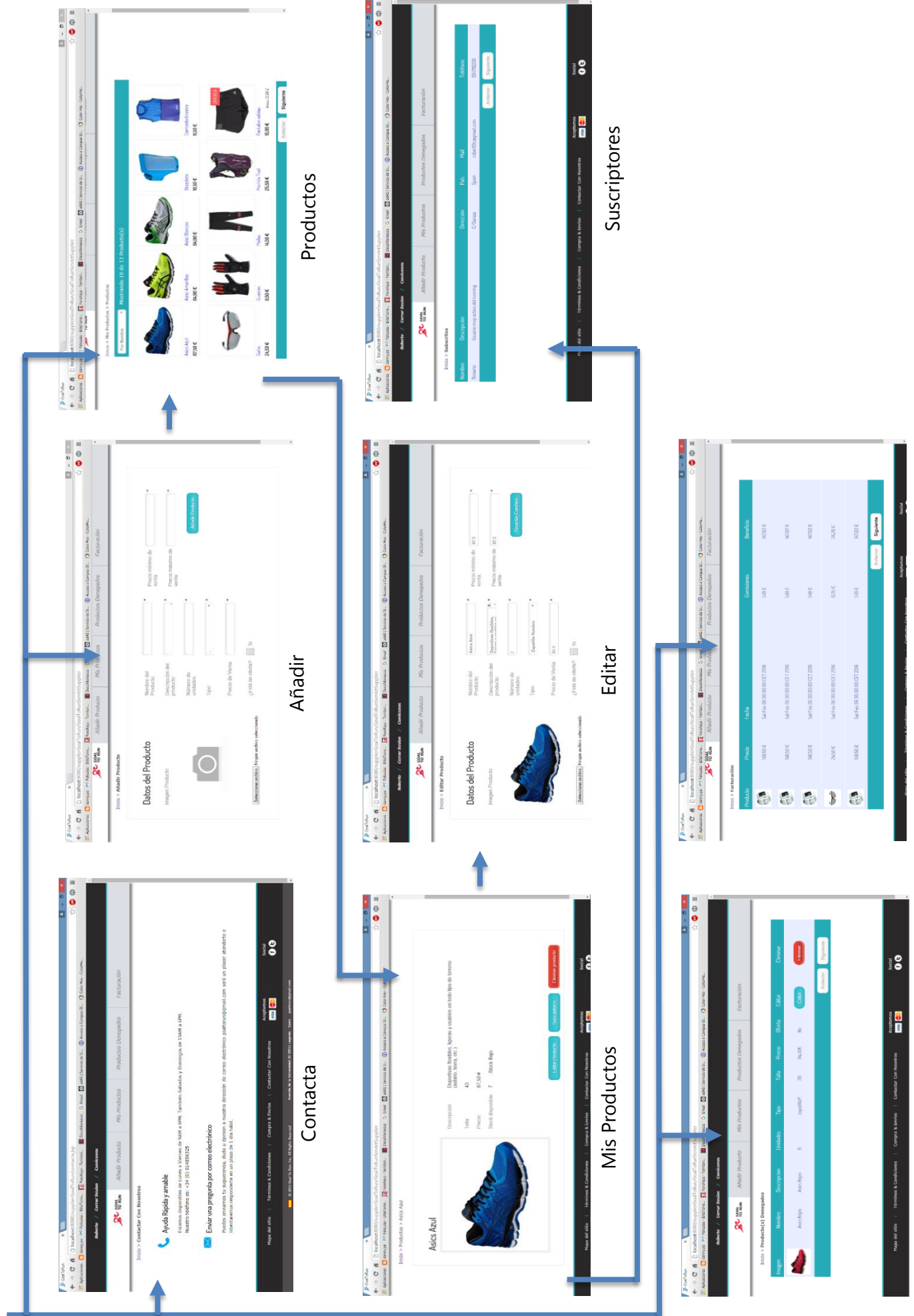


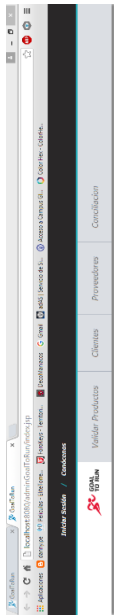
Ilustración 34 - Flujo Aplicación Web Proveedor

## Administrador

El administrador de la aplicación web tendrá a su disposición diferentes páginas, las cuales son descritas a continuación:

- **Clientes/Proveedores:** páginas que muestran información relativa a los usuarios o proveedores que hay registrados en la aplicación.
- **Conciliación:** página que realiza la conciliación de la empresa y de los proveedores.
- **Conócenos:** página con información sobre la empresa, para que el usuario nos conozca un poco más.
- **Contacta:** página que da información sobre cómo ponerse en contacto con la empresa.
- **Editar Cliente/Proveedor:** página para modificar la información de un cliente, si resulta dañina o se puede bloquear la cuenta si hizo algo mal.
- **Envíos:** página que contiene información relativa a los pedidos.
- **Index/login:** página de inicio de sesión de la aplicación.
- **Producto Rechazado:** a esta página se llega tras rechazar la validación de un producto, aquí el administrador deberá dar una explicación al proveedor de su denegación.
- **Términos y Condiciones:** página con información muy relevante, como por ejemplo las condiciones de uso, la privacidad, la propiedad intelectual, etc.
- **Validar Productos:** página para validar los productos que los proveedores quieren vender.



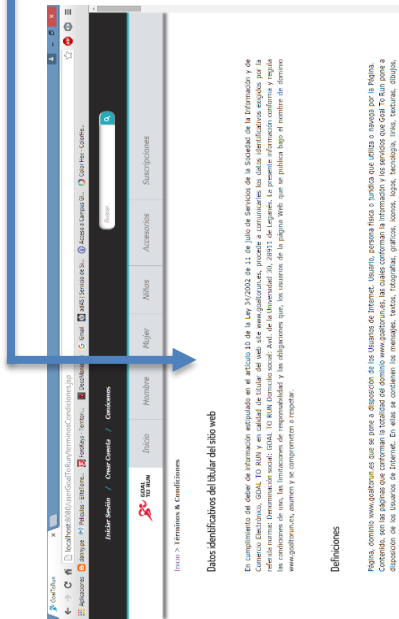


### Inicio Sesión

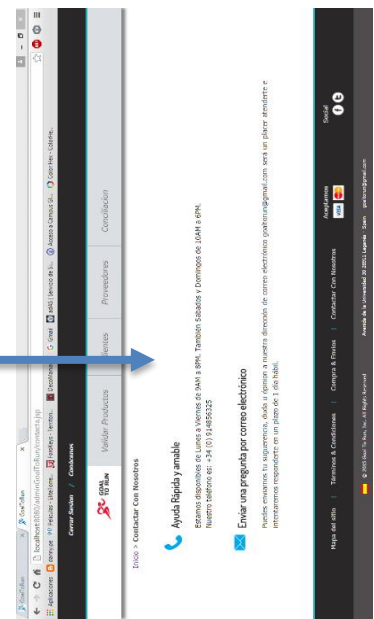
Email  
 Password



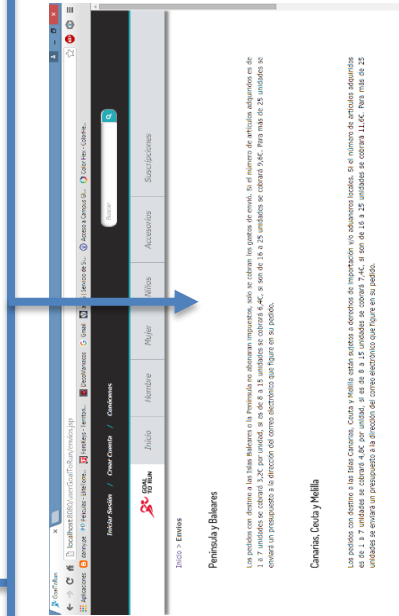
## Index/Login



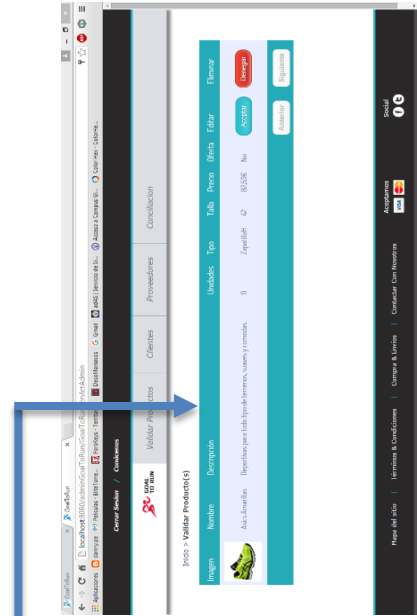
## Términos y Condiciones



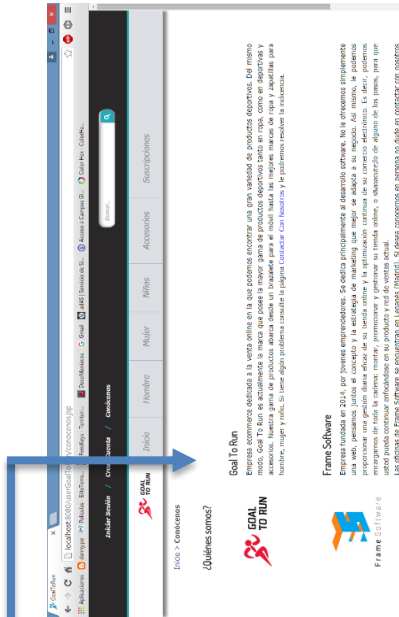
## Contacta



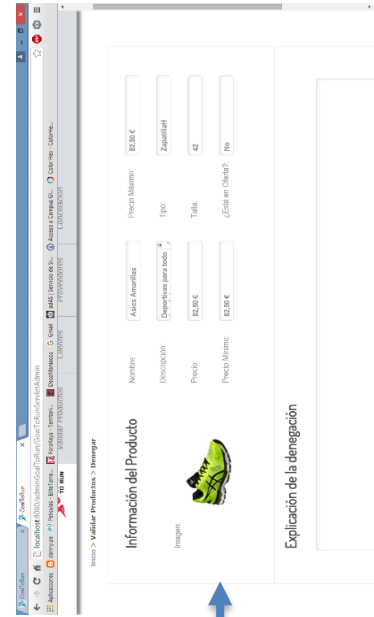
## Envíos



## Validar



## Conócenos



## Productos





## Servidor

En los siguientes apartados se mostrarán los pasos que se han seguido para una buena configuración del servidor y los diseños que se han realizado de las bases de datos.

### Configuración del Servidor

Una buena configuración es muy importante para que toda aplicación web funcione sin dar ningún problema. Por este motivo se van a explicar los pasos que se han seguido para configurar el servidor que se eligió en apartados anteriores *"Glassfish"*.

1. Cuando se procede a desplegar un proyecto web dinámico en Java, tenemos que configurar el siguiente fichero:
  - a. **Glassfish-web.xml:** fichero que recibe el nombre del servlet que tiene que desplegar para que la aplicación de usuario, proveedor o administrador funcione.
2. Como la aplicación web va a realizar conexiones de acceso a la Base de Datos, los pasos a seguir son:
  - a. Tener creada la Base de Datos.
  - b. Tener descargado y cargado el controlador JDBC de MySQL.
  - c. Entrar en el servidor en *"View Admin Console"*, ir al apartado *Resources* y entrar en *JDBC Connection Pools*. Una vez dentro hay que crear un conjunto de conexiones con las siguientes propiedades:
    - i. **Database Driver Vendor:** elegir el proveedor de la base de datos.
    - ii. **Initial and Minium Pool Size:** normalmente se inicializa con 8 conexiones a la base de datos, pero este valor puede ser cambiado.
    - iii. **Password:** establecer la contraseña para el usuario.
    - iv. **Pool\_Name:** nombre a elegir.
    - v. **Resource Type:** tipo de recurso a utilizar en la conexión.
    - vi. **URL:** normalmente sigue el siguiente formato.  
*"jdbc:mysql://localhost:3306/NombreBaseDatos"*
    - vii. **User:** establecer el usuario que desea tener acceso a la base de datos.  
Toda conexión va a utilizar el mismo usuario.

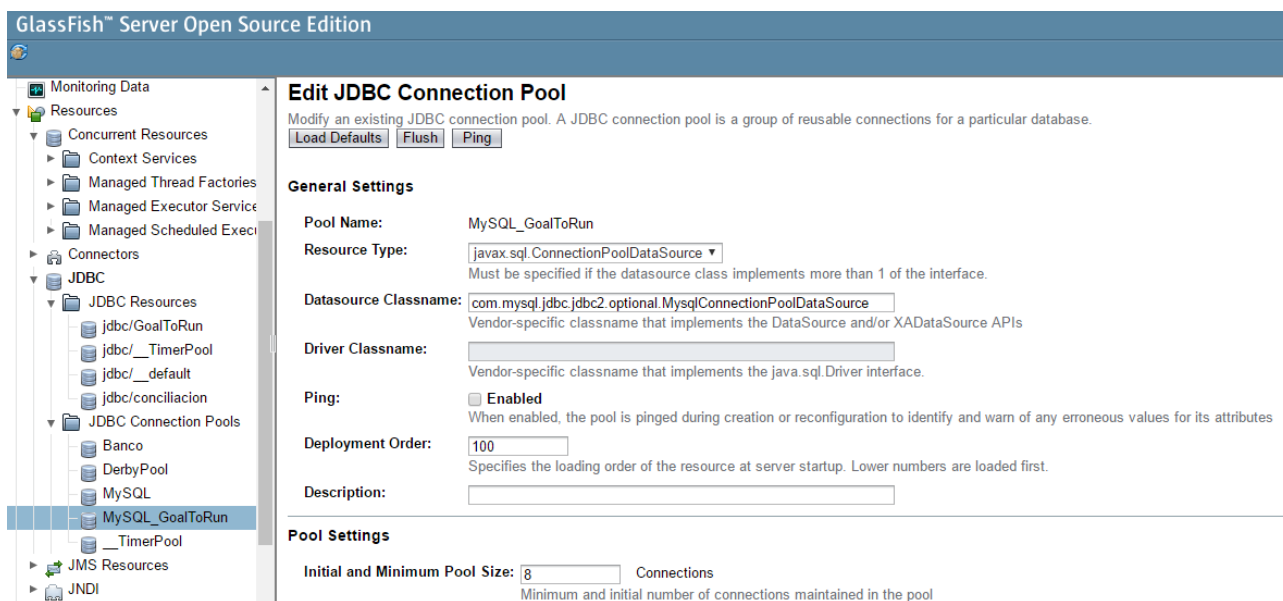


Ilustración 36 - JDBC Connection Pool

Add Property		Delete Properties
Name	Value	
URL	jdbc:mysql://localhost:3306/booreg	
user	yourUser	
password	yourPassword	

Ilustración 37 - Propiedades JDBC

- d. Tras guardar los datos, se procede a realizar un Ping para comprobar que se ha hecho bien todos los pasos anteriores. Seguimos adelante cuando nos informa que el Ping tuvo éxito.
- e. Por último, creamos el recurso JDBC que dará acceso a nuestro conjunto de conexiones. Para ello, vamos a *Resources* → *JDBC* → *JDBC Resources* y creamos uno nuevo, con estos parámetros:
  - i. **JNDI Name:** nombre a elegir.
  - ii. **Pool Name:** nombre que le dimos a nuestro conjunto de conexiones.

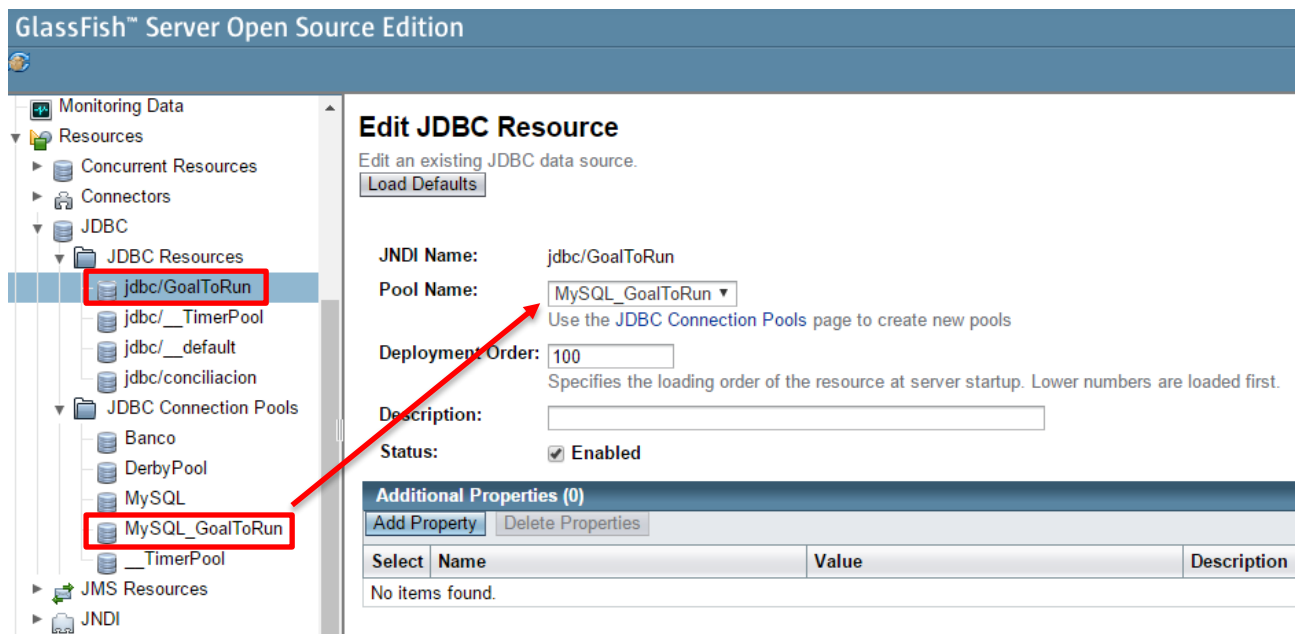


Ilustración 38 - JDBC Resource

3. Para poder comunicarnos dentro de la aplicación o para realizar al mismo tiempo varias acciones (a través de colas) necesitamos configurar *JMS*. Pasos a seguir:
  - a. Entrar en el servidor en “*View Admin Console*”, ir al apartado *JMS* y entrar en *JMS Connection Factory*. Una vez dentro hay que crear una fábrica de conexiones con las siguientes propiedades:
    - i. **Pool Name:** nombre a elegir.
    - ii. **Resource Type:** elegir un tipo de recurso para la fábrica de conexiones.
  - b. Tras guardar los datos, se procede a crear las conexiones que se estimen oportunas.

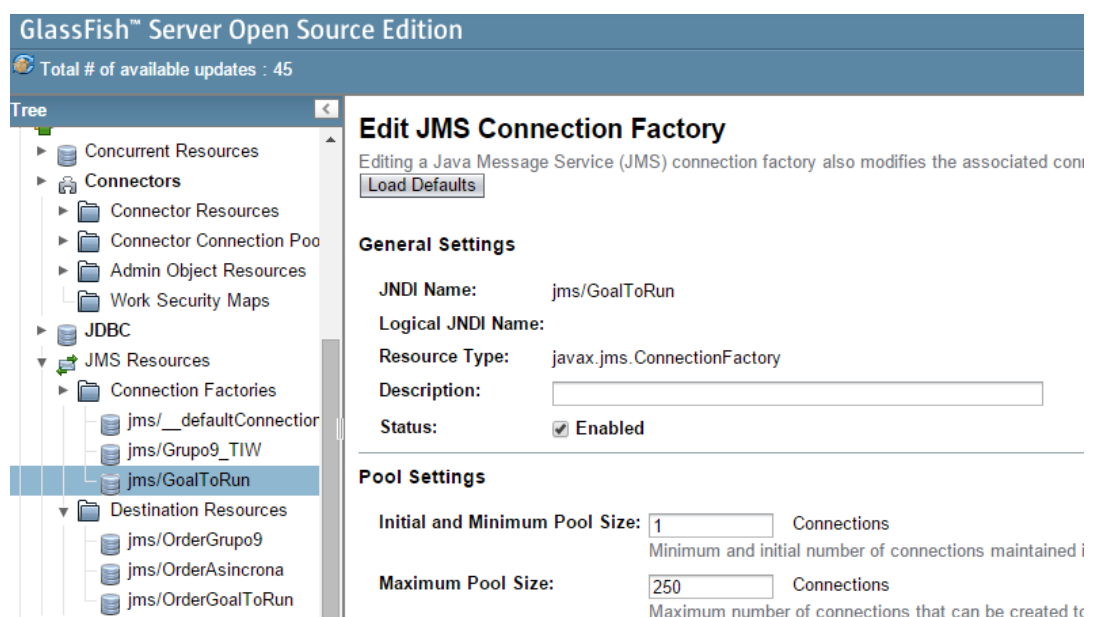


Ilustración 39 - JMS Connection Factory

## Base de Datos

Como se ha comentado en apartados anteriores, tendremos dos proyectos *JPA*. ¿Qué quiere decir esto? Pues bien, lo que quiere decir es que habrá dos creaciones de bases de datos. La primera que se explicará con detalle más adelante cuenta con siete tablas y su objetivo es la comunicación y acceso para toda la aplicación web. La segunda cuenta con dos tablas y su objetivo es servir de referencia al banco respecto a la empresa. Todos los elementos y relaciones que hay entre cada una de las tablas se podrán observar más adelante.

Ahora, se van a exponer los diferentes tipos que se usan para la definición de las columnas de la base de datos:

- ~ **Tipo DATE:** formato de fecha. Su representación es en formato de fecha numérica del tipo 'YYYY-MM-DD'. Su rango es '1000-01-01' hasta '9999-12-31'.
- ~ **Tipo ENUM** representa una enumeración. Puede tener un máximo de 65535 valores distintos. Es una cadena cuyo valor se elige de una lista enumerada de valores permitidos que se especifica al definir el campo. Puede ser una cadena vacía, incluso "null".
- ~ **Tipo FLOAT:** número de coma flotante de precisión simple. El rango de valores de precisión simple es de -3,402823466E+38 a -1,175494351E-38, 0, y 1,175494351E-38 hasta 3,402823466E+38.
- ~ **Tipo INT:** número entero con o sin signo. Con signo el rango de valores va desde -2147483648 a 2147483647. Sin signo, el rango va desde 0 a 4294967295.
- ~ **Tipo MEDIUMBLOB:** objeto binario que puede almacenar cualquier tipo de dato o información, acepta tamaños hasta 16.777.215 caracteres.
- ~ **Tipo TINYINT:** número entero con o sin signo. Con el rango de valores válidos desde -128 a 127. Sin signo, el rango de valores es de 0 a 255.
- ~ **Tipo VARCHAR:** almacena una cadena de longitud variable. La cadena podrá contener desde 0 a 255 caracteres.

Como se comentó anteriormente, en primer lugar se explica la base de datos general y justo después se explica la base de datos del banco.

## Base de Datos General

Esta base de datos se compone por 7 tablas:

1. **Users:** tabla que almacena los usuarios y proveedores registrados en el sistema, así como el administrador. Los campos con los que cuenta esta tabla son los siguientes:
  - a. **Address:** campo de la tabla para almacenar la dirección donde vive el usuario que se ha registrado.
  - b. **Address2:** campo de la tabla para almacenar el municipio donde vive el usuario que se ha registrado.
  - c. **City:** campo de la tabla para almacenar la ciudad donde vive el usuario que se ha registrado.
  - d. **Country:** campo de la tabla para almacenar el país donde vive el usuario que se ha registrado.
  - e. **Description:** campo para almacenar una descripción sobre el usuario que se ha registrado.
  - f. **Img:** campo para almacenar una imagen que identifique al usuario.
  - g. **Mail:** campo para almacenar el email de cada usuario registrado en el sistema.
  - h. **Name:** campo de la tabla para almacenar el nombre con el que el usuario se ha registrado.
  - i. **Password:** almacena la contraseña del usuario haciendo uso de una función de cifrado.
  - j. **Phone:** campo de la tabla para almacenar el teléfono que tiene el usuario que se ha registrado.
  - k. **Postal\_Code:** campo de la tabla para almacenar el código postal donde vive el usuario que se ha registrado.
  - l. **RutalImg:** campo para almacenar la ruta de la imagen.
  - m. **Surname:** campo de la tabla para almacenar el/los apellido/s con el que el usuario se ha registrado.
  - n. **Type:** campo para almacenar el tipo de usuario que se ha registrado.
2. **Produce:** tabla que almacena los productos añadidos por los proveedores. Los campos con los que cuenta esta tabla son los siguientes:

- a. **Description:** campo para almacenar la descripción del producto que se ha registrado.
  - b. **Discount:** campo que sirve para saber si un producto está o no en venta.
  - c. **IdProduct:** campo que almacena el id del producto que se registra.
  - d. **Img:** campo para almacenar una imagen que añada el proveedor del producto que ha registrado
  - e. **Max\_Price:** campo que almacena el precio máximo que tiene el producto que se ha registrado.
  - f. **Min\_Price:** campo que almacena el precio mínimo que tiene el producto que se ha registrado.
  - g. **Name:** campo para almacenar el nombre del producto que se ha registrado.
  - h. **Quantity:** campo que almacena la cantidad del producto registrado a vender.
  - i. **Rutalmg:** campo para almacenar la ruta de la imagen.
  - j. **Sale:** campo que sirve para saber si un producto se vendió o no.
  - k. **Sex:** campo que almacena el tipo de usuario que consume ese producto registrado (Hombre, Mujer o Niño).
  - l. **Size:** campo que almacena la talla del producto registrado. Hay productos que no requieren de talla.
  - m. **State:** campo que sirve para saber en qué estado está el producto registrado (0-Validando, 1-En venta, 2-Rechazado).
  - n. **Supplier:** campo que indica el email del proveedor que registra el producto.
  - o. **Type:** campo que almacena el tipo de producto que se registra.
3. **Order\_Product:** tabla que almacena los productos comprados por los usuarios. Los campos con los que cuenta esta tabla son los siguientes:
- a. **idOrder:** campo que almacena el id de compra.
  - b. **idProduct:** campo que almacena el id que identifica al producto
  - c. **Price:** campo que almacena el precio de la cantidad de productos comprados.

- d. **Quantity:** campo que almacena la cantidad del producto que se ha comprado.
- 4. **Discounts:** tabla que almacena todo lo que tiene que ver con los descuentos de la aplicación web. Los campos con los que cuenta esta tabla son los siguientes:
  - a. **Code:** campo que almacena el código descuento que tiene el usuario para usar la próxima vez que quiera.
  - b. **Expiration:** campo que almacena la fecha de expiración del código descuento.
  - c. **Used:** este campo sirve para saber si un vale ha sido usado o no.
- 5. **Billing:** tabla que los productos que se compran en la página. Los campos con los que cuenta esta tabla son los siguientes:
  - a. **Amount:** campo que almacena la cuantía total a pagar.
  - b. **Date:** campo que almacena la fecha en la que se produce la compra.
  - c. **idBilling:** campo que almacena el identificador de cada compra.
  - d. **idProduct:** campo que almacena el identificador del producto.
  - e. **Supplier:** campo que almacena el email del proveedor al que se le compra el producto.
- 6. **Subscription:** tabla que almacena las suscripciones de los usuarios a los productos de los proveedores. Los campos con los que cuenta esta tabla son los siguientes:
  - a. **idProduct:** campo que almacena el identificador del producto.
  - b. **idUser:** campo que almacena el identificador del usuario.

## Banco BBDD

Esta base de datos se compone por 2 tablas:

- 1. **ConciliacionEmpresa:** tabla que almacena la conciliación con la empresa. Los campos con los que cuenta esta tabla son los siguientes:
  - a. **Importe:** campo que almacena el importe de beneficio de la empresa.
  - b. **Month:** campo que almacena el mes en el que se produjo la conciliación.
  - c. **Year:** campo que almacena el año en el que se produjo la conciliación.

2. **ConciliacionBanco:** tabla que almacena la conciliación con el banco. Los campos con los que cuenta esta tabla son los siguientes:
- a. **CodigoProveedor:** campo que almacena el email del proveedor con el que realizo la conciliación.
  - b. **Importe:** campo que almacena beneficio del proveedor por la venta de sus productos.
  - c. **Month:** campo que almacena el mes en el que se produjo la conciliación.
  - d. **Year:** campo que almacena el año en el que se produjo la conciliación.



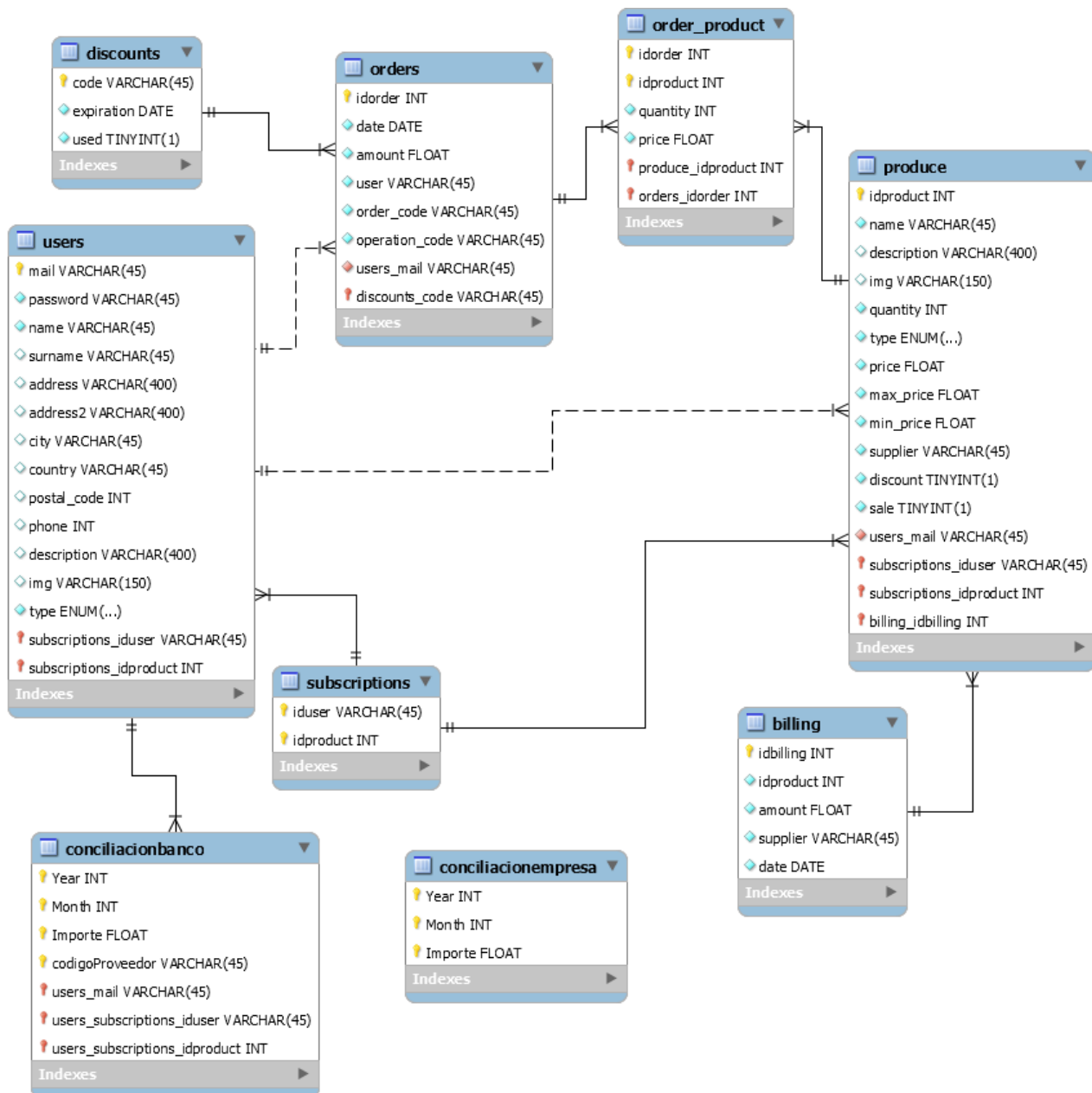


Ilustración 40 - Base de Datos Aplicación Web

## Capítulo V- Estrategia

En este capítulo se describe la metodología a seguir, el equipo de trabajo, las historias de usuario, es decir, los requisitos que el cliente desea para su aplicación y el sistema de pruebas para verificar el correcto funcionamiento de todo el conjunto.

### Selección de la metodología a usar

En el apartado "*Metodologías*" que se puede encontrar en el *Capítulo II – Estado del Arte*, se describen las dos metodologías de desarrollo de software más extendidas y usadas, como son la metodología en cascada y la metodología ágil. Ambas metodologías aportan de manera individual una serie de ventajas y desventajas que pueden ser elementos suficientes a la hora de decantarse por el uso de una en lugar de otra.

Entonces, ¿Qué metodología se va a usar? Pues bien, la elección de la metodología no ha sido realizada por las ventajas y desventajas de cada una, sino por la recomendación del tutor del proyecto, que se encuentra trabajando en el ámbito de las T.I. (Tecnologías de la Información). Su experiencia en el ámbito profesional de las metodologías ágiles y el poco uso que ha tenido a lo largo de toda la carrera, provocado por el mayor uso de la metodología en cascada, han supuesto motivos suficientes para desarrollar el proyecto con la metodología ágil Scrum.

### Scrum, la metodología ágil más extendida

Scrum es una metodología de gestión de proyectos ágiles en la que se aplican de manera regular, un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Es ampliamente utilizado por los equipos de desarrollo de software, de hecho, es la metodología ágil más extendida, ya que según el informe anual que publica VersionOne, el 72% de estos equipos usa Scrum o Scrum híbrido [32] [33].

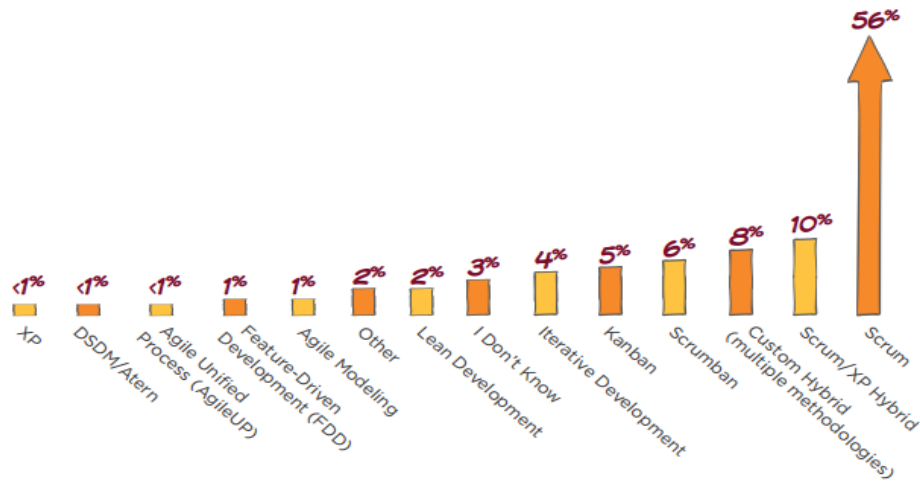


Ilustración 41 - Metodologías ágiles más usadas

Scrum se apoya en tres pilares fundamentales:

- ~ **Transparencia:** todo lo que afecta al resultado del trabajo debe ser conocido y visible para los responsables de los resultados. Un ejemplo de ello son las técnicas colaborativas que se usan para la comunicación entre los diferentes miembros del equipo.
- ~ **Inspección:** es necesario realizar un seguimiento de los artefactos y la evolución del trabajo para detectar desviaciones con respecto a lo previsto. Un método muy eficaz para realizar el control son las reuniones de equipo.
- ~ **Adaptación:** en el momento en el que se detecta alguna desviación o se tienen indicios de que esta se puede producir, es necesario actuar tan pronto como sea posible para minimizar al máximo la desviación y alcanzar los objetivos finales.

## Equipo de Trabajo

En la metodología ágil Scrum existen tres roles principales que son, el *Product Owner*, el *Scrum Master* y el equipo de desarrollo. En los siguientes apartados se define cada rol con más detalle.

### Product Owner

Representa la voz del cliente, es decir, es la persona que conoce lo que hay que desarrollar y el orden de desarrollo. Este rol es el responsable de gestionar las necesidades que serán satisfechas por el proyecto y asegurar el valor del trabajo que el equipo lleva a cabo [33].

El *Product Owner* es la persona responsable de la administración del *Backlog*. Esto incluye:

- ~ Definir claramente los elementos.
- ~ Priorizar los elementos en vista de lograr los objetivos y metas.
- ~ Optimizar el esfuerzo del Equipo de Desarrollo.

- ~ Asegurar la visibilidad, transparencia y claridad de los siguientes pasos para todo el equipo.
- ~ Verificar que cada miembro del equipo de desarrollo comprende los elementos del *Product Backlog* al nivel requerido.

El *Product Owner* puede hacer todas estas tareas o delegarlas en algún miembro del equipo. En cualquier caso, él será responsable de su correcta ejecución. Para lograr el éxito, se deben respetar todas sus decisiones.

## Scrum Master

El *Scrum Master* se encarga de hacer que las normas se cumplan, y por lo tanto, es el responsable de asegurarse que todo el equipo entiende Scrum y actúa en consecuencia siguiendo sus prácticas y reglas [33].

*Scrum* da soporte al *Product Owner* de varias maneras:

- ~ Le enseña técnicas para una gestión eficiente del *Product Backlog*.
- ~ Ayuda al Equipo *Scrum* a entender la necesidad de registrar los elementos del *Product Backlog* de forma concisa y clara.
- ~ Garantiza que el *Product Owner* pueda organizar el *Product Backlog* para maximizar su valor.
- ~ Enseña al *Product Owner* el concepto de agilidad.

El *Scrum Master* da soporte al equipo de desarrollo de varias maneras, incluyendo:

- ~ Capacitación para la auto-organización y la multifuncionalidad.
- ~ Crear productos de alto valor.
- ~ Entender los elementos del *Product Backlog*.
- ~ Facilitar la entrega de Eventos *Scrum*.
- ~ Eliminar los problemas que impide el avance del Equipo de Desarrollo.

## Equipo de desarrollo

Es el encargado de desarrollar el sistema. Está conformado por profesionales encargados de convertir las necesidades del *Product Owner* en un conjunto de funcionalidades del producto software final [33]. Define el crecimiento orgánico del producto.

Para que el equipo de desarrollo funcione de la manera correcta deberá auto-gestionarse de tal manera que:

- ~ Sea autónomo, es decir, la toma de decisiones se hace por el equipo en conjunto.
- ~ Sea adaptable, el equipo se ajusta de la mejor manera para resolver los problemas.
- ~ Sea responsable, todos los miembros del equipo comparten las responsabilidades de los resultados.

## Equipo de trabajo del proyecto

El equipo de trabajo del proyecto está compuesto por dos personas:

- ~ Jesús Hernando Corrochano, Tutor del proyecto, que actúa como *Scrum Master*.
- ~ Roberto Cañizares Sanz, que actúa como equipo de desarrollo auto-gestionable.
- ~ El papel de Product Owner es compartido entre Jesús Hernando Corrochano y Roberto Cañizares Sanz.

## Modelo de trabajo

Como se lleva comentado a lo largo del capítulo, la metodología que se va a usar en este proyecto será Scrum. Como cualquier otra metodología, Scrum también cuenta con su modelo de trabajo. Para hacerse una idea del modelo de trabajo que se ha realizado, la siguiente ilustración lo muestra a la perfección. En ella, se puede observar claramente los diferentes roles que intervienen a lo largo del proceso y que se describirán en apartados posteriores, además de los diferentes artefactos y eventos que también se dan cita en un proyecto Scrum.

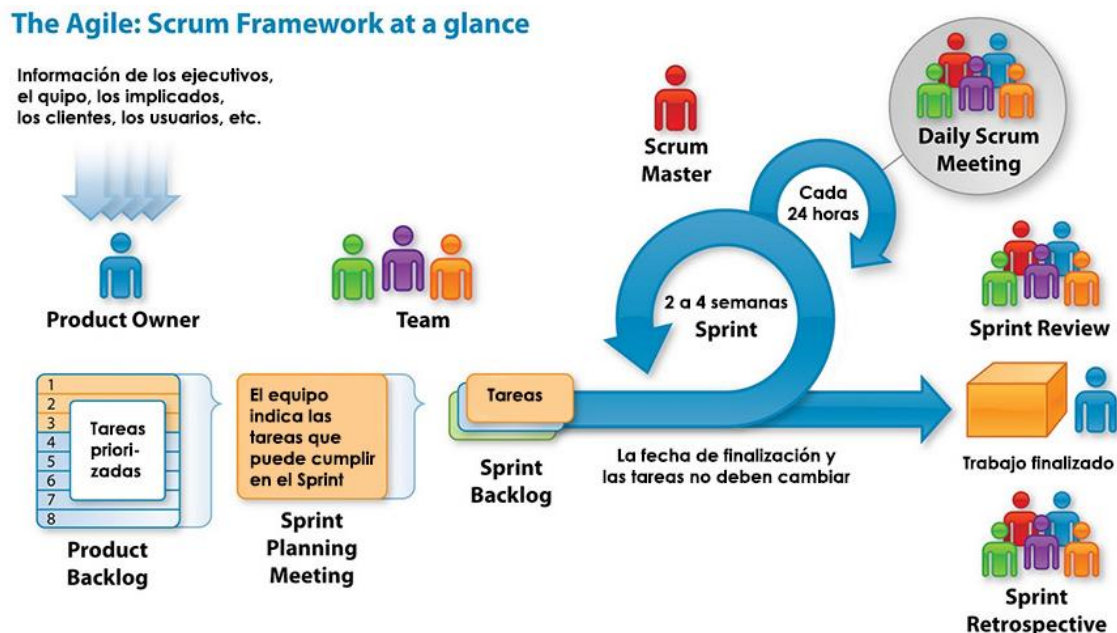


Ilustración 42 - Ciclo de vida de la metodología Scrum

## Eventos en Scrum

Los eventos en Scrum tienen como objetivo crear regularidad y minimizar la necesidad de reuniones no definidas en la metodología [34]. Todos los eventos deben tener una duración predefinida, de tal forma que el evento no pueda extenderse más allá de la duración programada. Los eventos que se pueden encontrar son:

- ~ **Sprint:** es el corazón de Scrum. Es un evento de tiempo fijo, de un mes (o menos) durante el cual se crea un incremento del producto que pasa a ser utilizable y potencialmente desplegable. Los *Sprints* más exitosos son aquellos que tienen una duración constante a lo largo de un desarrollo, por eso, la duración de cada *Sprint* debe ser fijada antes del comienzo del proyecto. Cada *Sprint* debe comenzar inmediatamente después de la finalización del *Sprint* previo y cada uno de ellos consiste en la Planificación del *Sprint* o "Planning Meeting", en los Scrums Diarios o "Daily Scrums", en el desarrollo, en sí, en la revisión del *Sprint* o "Sprint Review", en la Retrospectiva del *Sprint* o "Sprint Retrospective" y en las reuniones de *Grooming* o "Backlog Grooming". En el desarrollo del proyecto realizado, se ha planificado un total de cinco *Sprints* de una duración media de un mes (sin contar días festivos ni fines de semana).
- ~ **Sprint Planning Meeting:** es una reunión del equipo para planificar el trabajo a realizar en el *Sprint*. Con una duración máxima de ocho horas para un *Sprint* de un mes [34]. A lo largo del proyecto, se han realizado un total de tres reuniones presenciales tras finalizar cada *Sprint* entre el *Product Owner / Scrum Master* (Jesús Hernando Corrochano) y el *Product Owner /* equipo de desarrollo (Roberto Cañizares Sanz).

En la siguiente ilustración se puede observar los *Sprints* que se han llevado a cabo en este proyecto. Cada uno de estos *Sprints* se va a explicar en el *Capítulo VI – Gestión del Proyecto* en el apartado de "Planificación".



Ilustración 43 - Fechas de cada Sprints realizado en el proyecto

~ **Daily Scrum:** reunión diaria de 15 minutos sobre el estado del proyecto para que el equipo de desarrollo sincronice sus actividades y cree un plan para las siguientes 24 horas. Durante la reunión cada miembro del equipo de desarrollo tiene que contestar a tres preguntas [34].

1. ¿Qué hice ayer para ayudar al Equipo de Desarrollo a cumplir con el objetivo del *Sprint*?
2. ¿Qué voy a hacer hoy para ayudar al Equipo de Desarrollo a cumplir con el objetivo del *Sprint*?
3. ¿Identifico algún problema que me impida a mí o al Equipo de Desarrollo alcanzar el objetivo?

Realizar un *Scrum* Diario optimiza la probabilidad de que el Equipo de Desarrollo cumpla con el objetivo del *Sprint*.

~ **Sprint Review:** reunión informal del equipo completo junto con los interesados para inspeccionar lo realizado a lo largo del *Sprint*. Tiene una duración máxima de cuatro horas para un *Sprint* de un mes y el objetivo es:

1. Presentar el trabajo completado a los interesados.
2. Revisar el trabajo que fue completado y el que no lo fue.

Durante el desarrollo del proyecto, se han realizado un total de tres reuniones presenciales, una vez se finalizó cada *Sprint*.

~ **Sprint Retrospective:** la retrospectiva del *Sprint* tiene como objetivo realizar una reunión entre los miembros del equipo para compartir impresiones sobre el *Sprint* que acaban de terminar y para planificar el siguiente *Sprint*. Para los *Sprints* de un mes, esta reunión tiene un tiempo programado de tres horas. Los objetivos a cumplir son:

1. Revisar el desempeño del último *Sprint* con respecto a las personas, relaciones, procesos y herramientas.
2. Identificar y ordenar los principales aciertos y áreas de oportunidad
3. Crear un plan para la integración de mejoras.

A lo largo del proyecto desarrollado, se han realizado un total de tres reuniones presenciales, una vez se finalizó cada *Sprint*.

~ **Backlog Grooming:** reunión a la que suele asistir todo el equipo Scrum, incluyendo el *Product Owner* y *Scrum Master* para revisar el *Product Backlog* y añadir, retirar o reestimar las

historias de usuario. Es muy recomendable que se mantenga, al menos, una reunión de *grooming* cuando el *Sprint* esté llegando a su fin, de entre dos y cinco días antes para *Sprints* de un mes de duración, con el objetivo de mantener el *Backlog* actualizado y optimizar el valor de sus ítems. En resumen, estas reuniones que han sido tanto presenciales como llevadas por Skype han servido para preparar el siguiente *Sprint*.

## Artefactos en Scrum

Los artefactos de Scrum sirven para representar trabajo de tal forma que el equipo completo de Scrum tenga una visión transparente del proyecto. Los artefactos son los siguientes:

- ~ **Product Backlog:** es un listado ordenado y priorizado de todo lo que se quiere añadir al producto, donde generalmente las historias de usuario son el elemento más común en la lista. Es dinámico porque está en constante evolución debido al producto y al entorno, lo que le hace mantenerse competitivo y útil. El responsable del *Product Backlog* es el *Product Owner*, el cual se encarga de alimentar, ordenar y mantener actualizado el listado.

En el proyecto desarrollado, el *Product Backlog* se ha creado mediante un listado de las diferentes historias de usuario que se han añadido al proyecto. Se ha hecho uso de la herramienta colaborativa Trello [35].

- ~ **Sprint Backlog:** es un conjunto de elementos que han sido seleccionados del *Product Backlog* para desarrollar el *Sprint* actual. A medida que el trabajo se completa se va actualizando la lista de elementos. Su objetivo es hacer visible todo el trabajo que el Equipo de Desarrollo ha identificado como necesario para cumplir el *Sprint*.

Para el listado de elementos que se han ido desarrollando a lo largo de cada *Sprint*, también se ha hecho uso de la herramienta Trello [35].

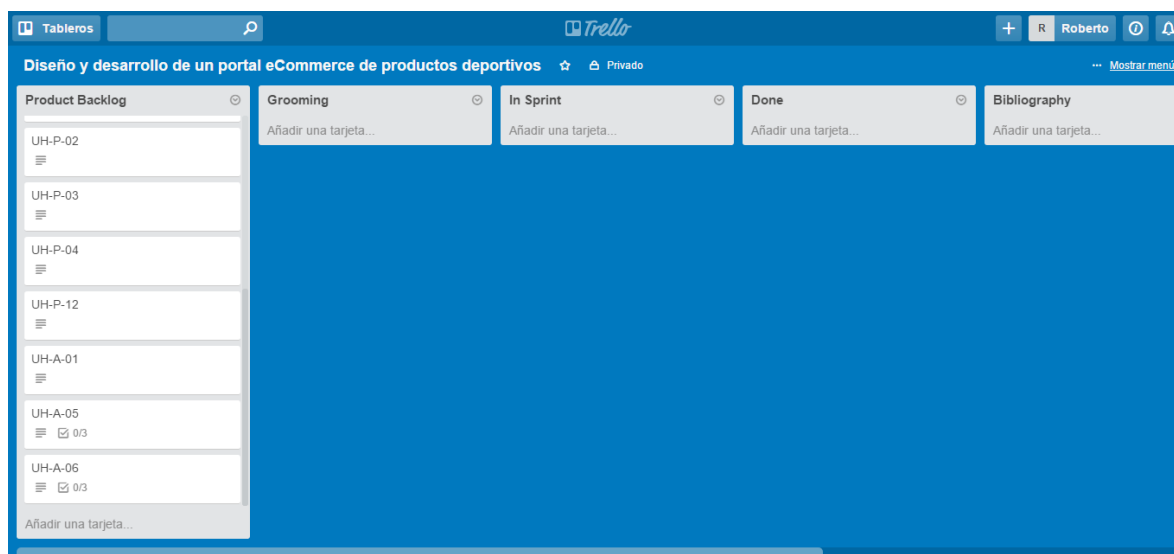


Ilustración 44 - Product Backlog del proyecto en Trello



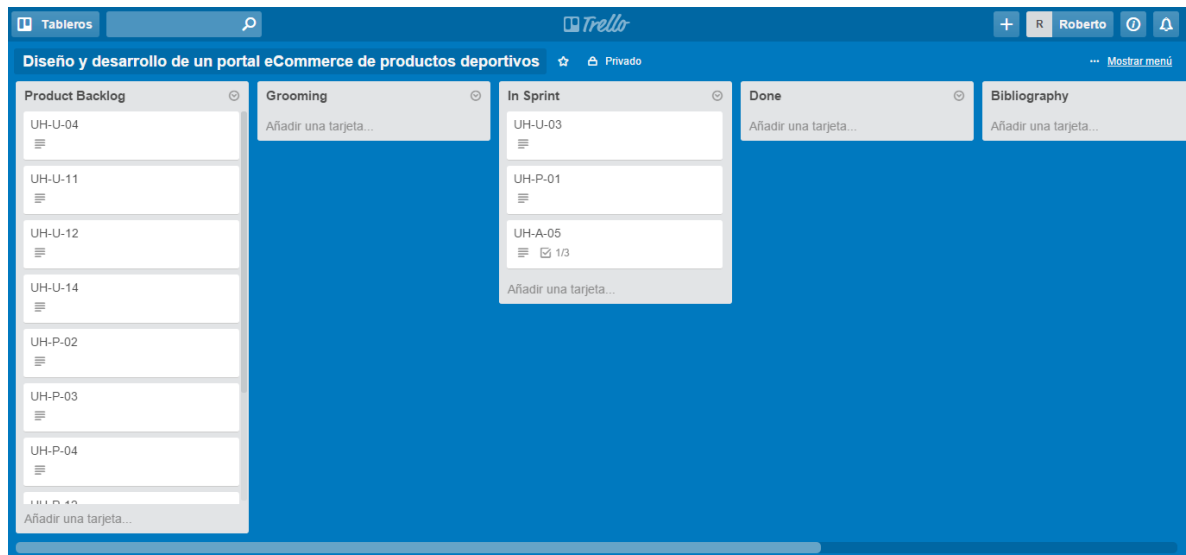


Ilustración 45 - Ejemplo de Sprint en Trello

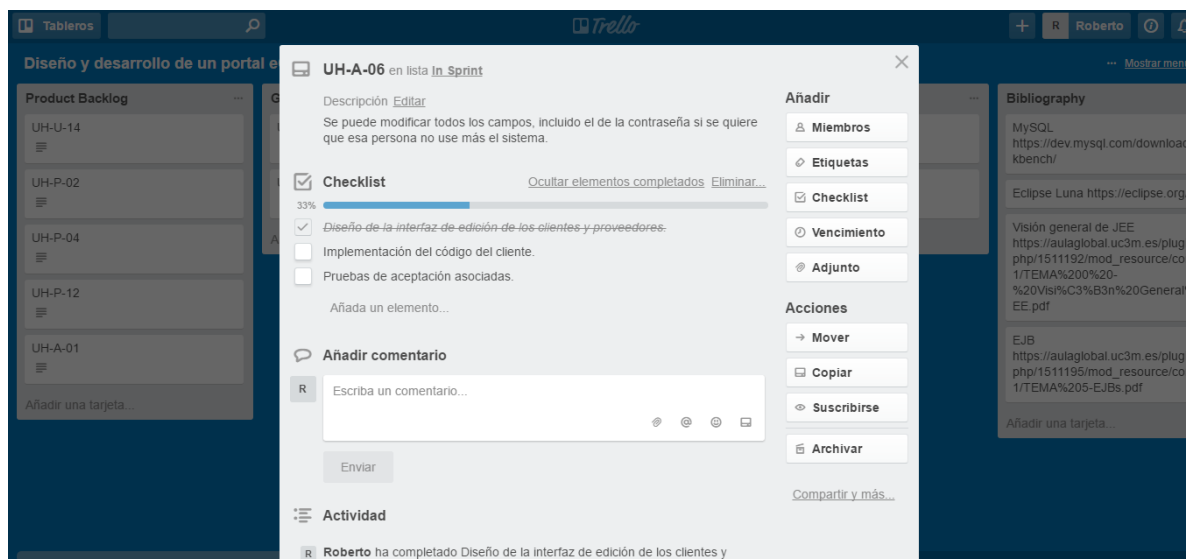


Ilustración 46 - Ejemplo de un Historia de Usuario en Trello

~ **Ciclo de vida iterativo e incremental:** corresponde con la suma de todos los elementos del *Product Backlog* completados durante un *Sprint* y los *Sprints* anteriores.

Dónde:

- ~ **Product Backlog:** listado ordenado de historias de usuario aún no implementadas ni pertenecientes a ningún *Sprint*.
- ~ **Grooming:** listado de historias de usuario que se podrán realizar en el siguiente *Sprint*.
- ~ **In Sprint:** listado de historias de usuario del *Sprint* actual.
- ~ **Done:** listado de historias de usuario completadas.
- ~ **Bibliography:** bibliografía a la que se ha hecho referencia.

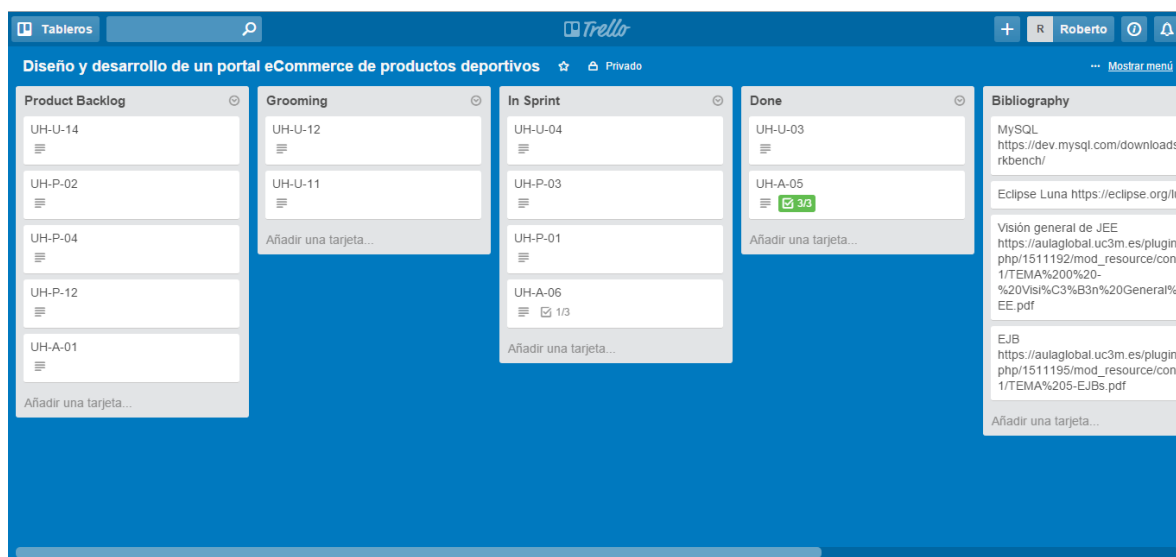


Ilustración 47 - Ciclo de vida del Sprint 1 en Trello

## Filosofía de trabajo

La filosofía de trabajo seguida a lo largo de un proyecto puede suponer el éxito o fracaso del mismo. En el proyecto realizado, se ha seguido una filosofía de trabajo que incluye el uso de herramientas colaborativas, reuniones presenciales y el uso de aplicaciones de copia de seguridad.

### Reuniones presenciales del equipo de trabajo

Las reuniones presenciales que se han realizado entre los miembros del proyecto han tenido lugar en el despacho del tutor. También se hizo uso de las instalaciones donde trabaja el tutor para realizar las pruebas oportunas al producto.

### Reuniones no presenciales del equipo de trabajo

Las reuniones no presenciales se han llevado a cabo a través de videollamadas con la herramienta Skype [36].

### Herramientas colaborativas

Se ha hecho uso de la herramienta colaborativa Trello [35] para gestionar la planificación. En la siguiente ilustración se puede observar el tablero correspondiente al proyecto desarrollado, donde aparecen los siguientes elementos:

- ~ *Product Backlog*: listado ordenado de historias de usuario aún no implementadas ni pertenecientes a ningún *Sprint*.
- ~ *Grooming*: listado de historias de usuario que se podrán realizar en el siguiente *Sprint*.
- ~ *In Sprint*: listado de historias de usuario del *Sprint* actual.

- ~ Done: listado de historias de usuario completadas.
- ~ Bibliography: bibliografía que se ha hecho referencia.

## Repositorio de código fuente

Para evitar una posible pérdida del trabajo diario que se está realizando, se ha usado un repositorio privado para almacenar el código fuente, la base de datos, su modelo y la memoria del trabajo. El sistema de almacenamiento empleado ha sido Dropbox [37]. También se han gestionado las versiones internas en base a las historias de usuario que se iban completando. El repositorio del proyecto creado para el cliente contiene:

- ~ Los proyectos *EAR (Enterprise Application Project)* correspondientes para el funcionamiento completo de la aplicación.
- ~ Los ficheros SQL necesarios para crear las bases de datos y su modelo.
- ~ La memoria del trabajo.

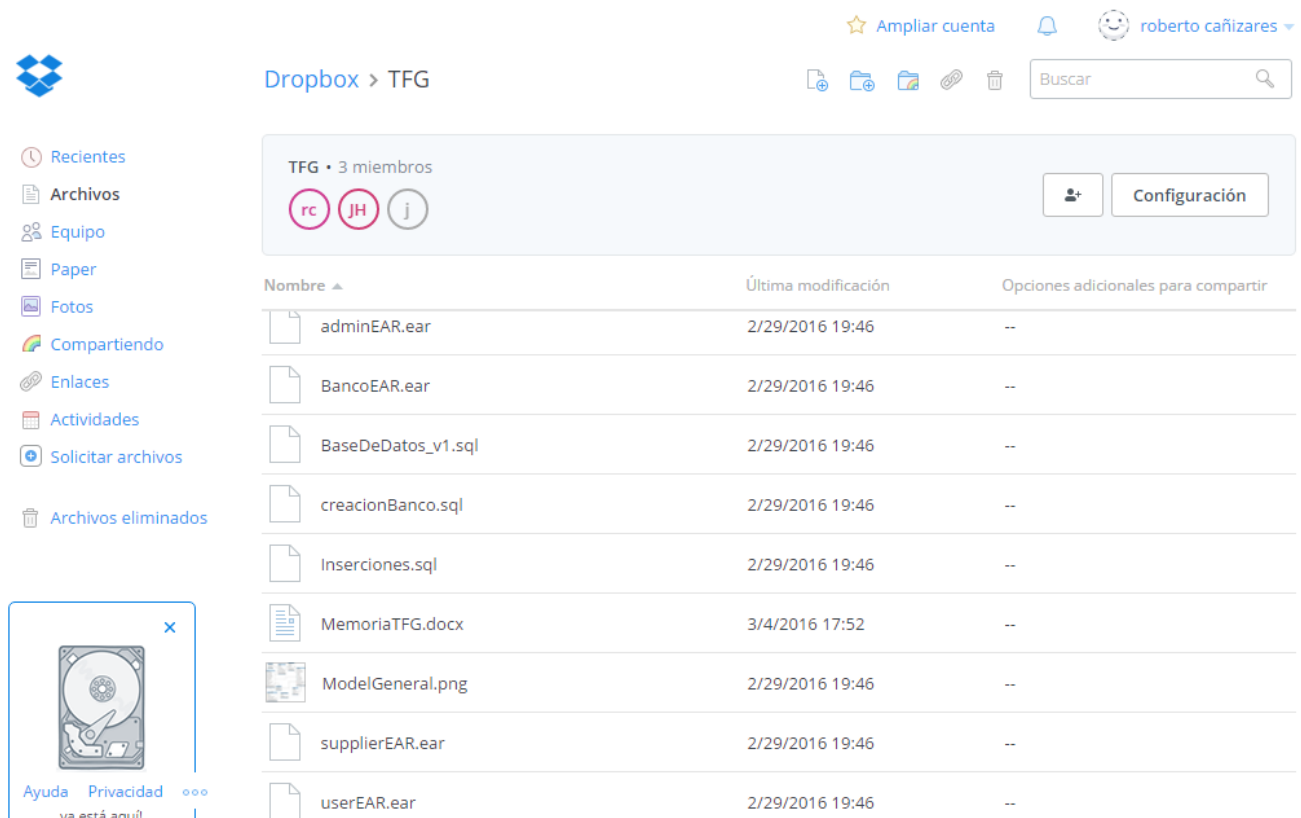


Ilustración 48 - Repositorio de código fuente en Dropbox

## Historias de Usuario

En la metodología ágil Scrum no se utilizan las especificaciones de requisitos para definir el comportamiento que tendrá la aplicación. En su lugar, se utilizan las historias de usuario que tienen su origen en la metodología "eXtreme Programming" o programación extrema [33]. Las historias de usuario muestran la funcionalidad que va a ser desarrollado, pero no cómo se va a desarrollar.

Ron Jeffries [38] escribía que una historia de usuario no es solo una descripción de una funcionalidad, sino que está formada por:

- a) **Creación de la tarjeta:** debe contener como mínimo una descripción escrita con un título asociado que sirve como valor de identificación de la funcionalidad. También puede contener la estimación y el valor de negocio que tiene la historia para el cliente.
- b) **Conversación:** corresponde al dialogo que se ha llevado a cabo entre el cliente y el usuario para aclarar los detalles y las dudas que puedan surgir de la historia de usuario.
- c) **Confirmación:** en esta última parte el equipo del proyecto y el cliente seleccionarán las pruebas que se van a llevar a cabo para comprobar que la historia se ha completado con éxito.

En las historias de usuario que se han escrito para la realización de este proyecto se han tenido en cuenta las especificaciones descritas por Ron Jeffries. De esta manera las historias que se muestran a continuación están formadas por:

- ~ **Identificador:** identificador unívoco de la historia.
- ~ **Título:** título descriptivo de la historia.
- ~ **Valor:** valor que aporta la historia de usuario para el cliente. Cuanto mayor valor posea mayor prioridad tendrá la historia de usuario. Este valor está descrito por el cliente y el equipo de desarrollo. Junto con el campo estimación, que va a determinar el orden en el que las historias de usuario serán implementadas.
- ~ **Estimación:** estimación del tiempo de implementación de la historia de usuario. Para estimar el tiempo se ha usado una medida denominada en métodos ágiles como "puntos historia", que se trata de una fusión de la cantidad de esfuerzo que supone desarrollar la historia de usuario, la complejidad de su desarrollo y el riesgo inherente. La medida que será usada se corresponde con la sucesión de Fibonacci y estará limitada a 7 valores, siendo la medida a usar "1, 2, 3, 5, 8, 13, 21" y que corresponderá con el tiempo teórico de desarrollo por persona en cada día de trabajo, de tal manera que 1 corresponde a un día de trabajo y 21 corresponde

a 21 días de trabajo. Se suponen jornadas laborales de 6 horas diarias. Se ha usado la serie de Fibonacci ya que la separación entre los números crece a medida que los números aumentan. Es muy útil para estimar grandes unidades de trabajo [33].

- ~ **Dependencias:** identificador de las historias de usuario dependientes de la misma.
- ~ **Descripción:** descripción sintetizada de la historia de usuario. Se seguirá el patrón descrito por Mike Cohn [39] "Como [rol del beneficiario], quiere [objetivo], para poder [beneficio]", para garantizar que la funcionalidad capturada será de alto nivel y que la extensión no sea demasiado larga.
- ~ **Tareas:** descripción de las tareas que hay que realizar para completar la historia de usuario.
- ~ **Pruebas de aceptación:** pruebas consensuadas entre el cliente y el equipo de desarrollo que el código debe superar para dar por finalizada una historia de usuario. Se referencia a la prueba o pruebas que la historia de usuario debe pasar. Las pruebas se van a definir en el siguiente apartado.
- ~ **Confirmaciones:** dialogo que se ha seguido entre el equipo y el cliente para aclarar los detalles y las dudas que pudieran surgir de la historia de usuario.

Para definir las historias de usuario de una manera más cómoda y más entendible, se van a mostrar en relación a cada una de las partes fundamentales de la aplicación (usuario, proveedor y administrador). Quedan recogidas de la siguiente manera:

## Usuario

UH-U-01			
<b>IDENTIFICADOR:</b>	UH-U-01	<b>TÍTULO:</b>	Visualización de los productos
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	13
<b>DEPENDENCIAS:</b>	-	<b>DESCRIPCIÓN:</b>	Como cliente, quiero visualizar todos los productos sin la necesidad de registrarme para poder decidir si compro alguno o no.
<b>CONFIRMACIONES:</b>			
~ Los productos se deben mostrar por sexo de la persona y por el tipo de artículo. ~ Se mostrarán un total de 10 productos por página. ~ Se pondrá el nombre, el precio actual y el anterior si está en oferta. ~ Se deben añadir los botones de "anterior" y "siguiente" para buscar más productos.			
<b>TAREAS:</b>			
~ Diseño de la interfaz de productos. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-06 ~ TEST-07			

Tabla 1 - Historia de Usuario 1. UH-U-01

UH-U-02			
<b>IDENTIFICADOR:</b>	UH-U-02	<b>TÍTULO:</b>	Visualizar un producto
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	5
<b>DEPENDENCIAS:</b>	UH-U-01	<b>DESCRIPCIÓN:</b>	Como cliente, quiero visualizar toda la información acerca de un producto para poder decidirme si lo compro.
<b>CONFIRMACIONES:</b>			
~ Se mostrará la imagen del producto. ~ Se mostrará el nombre, la descripción, el precio, la talla y el stock. ~ Se podrá suscribir a él o comprarlo, (si el cliente inicio sesión).			
<b>TAREAS:</b>			
~ Diseño de la interfaz del producto. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-08			

Tabla 2 - Historia de Usuario 2. UH-U-02

UH-U-03			
<b>IDENTIFICADOR:</b>	UH-U-03	<b>TÍTULO:</b>	Registro
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	2
<b>DEPENDENCIAS:</b>	UH-U-04	<b>DESCRIPCIÓN:</b>	Como cliente, quiero poder crearme una cuenta para poder suscribirme o comprar productos.
<b>CONFIRMACIONES:</b>			
~ Cuando el cliente se registra, sus datos quedan almacenados en la base de datos.			
<b>TAREAS:</b>			
~ Diseño de la interfaz del registro. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-03			

Tabla 3 - Historia de Usuario 3. UH-U-03

UH-U-04			
<b>IDENTIFICADOR:</b>	UH-U-04	<b>TÍTULO:</b>	Inicio de sesión
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	2
<b>DEPENDENCIAS:</b>	UH-U-03	<b>DESCRIPCIÓN:</b>	Como cliente, quiero iniciar o cerrar sesión, para poder acceder o salir del sistema respectivamente.
<b>CONFIRMACIONES:</b>			
~ Al iniciar sesión, se conservarán los datos y se redirigirá a la página principal. ~ Al cerrar sesión, los datos de esa sesión se eliminarán.			
<b>TAREAS:</b>			
~ Diseño de la interfaz del inicio de sesión. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-01 ~ TEST-02			

Tabla 4 - Historia de Usuario 4. UH-U-04

UH-U-05			
<b>IDENTIFICADOR:</b>	UH-U-05	<b>TÍTULO:</b>	Suscripción a un producto
<b>VALOR:</b>	80	<b>ESTIMACIÓN:</b>	3
<b>DEPENDENCIAS:</b>	UH-U-03 y UH-U-04	<b>DESCRIPCIÓN:</b>	Como cliente, quiero poder suscribirme a un producto para poder recibir información de él.
<b>CONFIRMACIONES:</b>			
~ El cliente debe iniciar sesión para poder suscribirse. ~ Tras suscribirse al producto, será redirigido a la página de suscripciones. ~ Se puede dejar de estar suscrito, si se elimina la suscripción.			
<b>TAREAS:</b>			
~ Diseño de la interfaz de suscripciones. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-10			

Tabla 5 - Historia de Usuario 5. UH-U-05

UH-U-06			
<b>IDENTIFICADOR:</b>	UH-U-06	<b>TÍTULO:</b>	Carrito
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	5
<b>DEPENDENCIAS:</b>	UH-U-02, UH-U-03 y UH-U-04	<b>DESCRIPCIÓN:</b>	Como cliente, quiero ver mi carrito para poder ver los productos que voy a comprar.
<b>CONFIRMACIONES:</b>			
~ Para añadir productos al carrito se debe iniciar sesión. ~ Se puede visualizar información acerca de los productos (Nombre, Descripción y Precio). ~ Se puede cancelar el producto si no se desea comprar. ~ Se puede ver la cantidad total a pagar.			
<b>TAREAS:</b>			
~ Diseño de la interfaz del carrito. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-11			

Tabla 6 - Historia de Usuario 6. UH-U-06



UH-U-07			
<b>IDENTIFICADOR:</b>	UH-U-07	<b>TÍTULO:</b>	Compra
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	8
<b>DEPENDENCIAS:</b>	UH-U-03, UH-U-04 y UH-U-06	<b>DESCRIPCIÓN:</b>	Como cliente, quiero verificar mis datos de compra para poder quedarme tranquilo de que la compra se realiza correctamente.
<b>CONFIRMACIONES:</b>			
~ Se puede verificar los datos personales y de envío. ~ Se puede añadir vales descuentos.			
<b>TAREAS:</b>			
~ Diseño de la interfaz de compra. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-12			

Tabla 7 - Historia de Usuario 7. UH-U-07

UH-U-08			
<b>IDENTIFICADOR:</b>	UH-U-08	<b>TÍTULO:</b>	Confirmación de Compra
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	5
<b>DEPENDENCIAS:</b>	UH-U-03, UH-U-04 y UH-U-07	<b>DESCRIPCIÓN:</b>	Como cliente, quiero una confirmación de la compra para poder saber que se realizó con éxito.
<b>CONFIRMACIONES:</b>			
~ Se puede verificar los datos introducidos en los pasos anteriores.			
<b>TAREAS:</b>			
~ Diseño de la interfaz de confirmación de compra. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-13			

Tabla 8 - Historia de Usuario 8. UH-U-08

UH-U-09			
<b>IDENTIFICADOR:</b>	UH-U-09	<b>TÍTULO:</b>	Resumen
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	5
<b>DEPENDENCIAS:</b>	UH-U-03, UH-U-04 y UH-U-08	<b>DESCRIPCIÓN:</b>	Como cliente, quiero obtener un resumen de la compra y códigos descuentos para poder seguir confiando y comprando en la web.
<b>CONFIRMACIONES:</b>			
~ Se puede ver el código de la compra. ~ Se pueden ver los vales promocionales si la compra supera una cierta cantidad de dinero, y hasta qué fecha son válidos.			
<b>TAREAS:</b>			
~ Diseño de la interfaz de resumen. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-14			

Tabla 9 - Historia de Usuario 9. UH-U-09

UH-U-10			
<b>IDENTIFICADOR:</b>	UH-U-10	<b>TÍTULO:</b>	Mail
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	8
<b>DEPENDENCIAS:</b>	UH-U-03, UH-U-04 y UH-U-09	<b>DESCRIPCIÓN:</b>	Como cliente, quiero que me llegue un mensaje con la compra que acabo de realizar al correo para poder verificarla.
<b>CONFIRMACIONES:</b>			
~ Solo funciona con cuentas Gmail. ~ En el PDF, se puede visualizar los productos comprados con su información respectiva. También se puede visualizar el código de compra, el vale descuento y el de expedición.			
<b>TAREAS:</b>			
~ Diseño del mensaje. ~ Diseño del PDF. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-15			

Tabla 10 - Historia de Usuario 10. UH-U-10

UH-U-11			
<b>IDENTIFICADOR:</b>	UH-U-11	<b>TÍTULO:</b>	Ver Perfil
<b>VALOR:</b>	80	<b>ESTIMACIÓN:</b>	2
<b>DEPENDENCIAS:</b>	UH-U-03 y UH-U-04	<b>DESCRIPCIÓN:</b>	Como cliente, quiero ver mi perfil para poder comprobar que mis datos sean correctos.
<b>CONFIRMACIONES:</b>			
~ Se pueden visualizar todos los datos almacenados del usuario, excepto la contraseña.			
<b>TAREAS:</b>			
~ Diseño de la interfaz del perfil. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-04			

Tabla 11 - Historia de Usuario 11. UH-U-11

UH-U-12			
<b>IDENTIFICADOR:</b>	UH-U-12	<b>TÍTULO:</b>	Editar Perfil
<b>VALOR:</b>	80	<b>ESTIMACIÓN:</b>	2
<b>DEPENDENCIAS:</b>	UH-U-11	<b>DESCRIPCIÓN:</b>	Como cliente, quiero acceder a la edición de mi perfil para poder modificar mis datos.
<b>CONFIRMACIONES:</b>			
~ Se pueden modificar los datos almacenados del cliente.			
<b>TAREAS:</b>			
~ Diseño de la interfaz de edición del perfil. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-05			

Tabla 12 - Historia de Usuario 12. UH-U-12

UH-U-13			
<b>IDENTIFICADOR:</b>	UH-U-13	<b>TÍTULO:</b>	Histórico
<b>VALOR:</b>	80	<b>ESTIMACIÓN:</b>	3
<b>DEPENDENCIAS:</b>	UH-U-11	<b>DESCRIPCIÓN:</b>	Como cliente, quiero ver las compras que he realizado en la aplicación para poder saber qué productos suelo comprar más y cuánto dinero llevo gastado.
<b>CONFIRMACIONES:</b>			
~ Se pueden ver los productos comprados y el precio pagado por ellos. ~ Se puede visualizar el dinero gastado en total.			
<b>TAREAS:</b>			
~ Diseño de la interfaz del Histórico. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-16			

Tabla 13 - Historia de Usuario 13. UH-U-13

UH-U-14			
<b>IDENTIFICADOR:</b>	UH-U-14	<b>TÍTULO:</b>	Contraseña
<b>VALOR:</b>	90	<b>ESTIMACIÓN:</b>	5
<b>DEPENDENCIAS:</b>	-	<b>DESCRIPCIÓN:</b>	Como cliente, quiero recuperar mi contraseña en caso de olvido para poder iniciar sesión y garantizar la seguridad de los datos.
<b>CONFIRMACIONES:</b>			
~ Se enviará una nueva contraseña al cliente y luego él la podrá modificar.			
<b>TAREAS:</b>			
~ Diseño de la interfaz de recuperación de contraseña. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-09			

Tabla 14 - Historia de Usuario 14. UH-U-14

## Proveedor

UH-P-01			
<b>IDENTIFICADOR:</b>	UH-P-01	<b>TÍTULO:</b>	Registro
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	2
<b>DEPENDENCIAS:</b>	UH-P-02	<b>DESCRIPCIÓN:</b>	Como proveedor, quiero poder crear una cuenta para poder vender mis productos.
<b>CONFIRMACIONES:</b>			
~ Cuando el proveedor se registra, sus datos quedan almacenados en la base de datos.			
<b>TAREAS:</b>			
~ Diseño de la interfaz del registro. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-03			

Tabla 15 - Historia de Usuario 14. UH-P-01

UH-P-02			
<b>IDENTIFICADOR:</b>	UH-P-02	<b>TÍTULO:</b>	Inicio de sesión
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	2
<b>DEPENDENCIAS:</b>	UH-P-01	<b>DESCRIPCIÓN:</b>	Como proveedor, quiero iniciar o cerrar sesión, para poder acceder o salir del sistema respectivamente.
<b>CONFIRMACIONES:</b>			
~ Al iniciar sesión, se conservarán los datos y se redirigirá a la página principal. ~ Al cerrar sesión, los datos de esa sesión se eliminarán.			
<b>TAREAS:</b>			
~ Diseño de la interfaz del inicio de sesión. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-01 ~ TEST-02			

Tabla 16 - Historia de Usuario 15. UH-P-02

UH-P-03			
<b>IDENTIFICADOR:</b>	UH-P-03	<b>TÍTULO:</b>	Ver Perfil
<b>VALOR:</b>	80	<b>ESTIMACIÓN:</b>	2
<b>DEPENDENCIAS:</b>	UH-P-01 y UH-P-02	<b>DESCRIPCIÓN:</b>	Como proveedor, quiero ver mi perfil para poder comprobar que mis datos sean correctos.
<b>CONFIRMACIONES:</b>			
~ Se puede visualizar todos los datos almacenados del proveedor, excepto la contraseña.			
<b>TAREAS:</b>			
~ Diseño de la interfaz del perfil. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-04			

Tabla 17 - Historia de Usuario 16. UH-P-03

UH-P-04			
<b>IDENTIFICADOR:</b>	UH-P-04	<b>TÍTULO:</b>	Editar Perfil
<b>VALOR:</b>	80	<b>ESTIMACIÓN:</b>	2
<b>DEPENDENCIAS:</b>	UH-P-03	<b>DESCRIPCIÓN:</b>	Como proveedor, quiero acceder a la edición de mi perfil para poder modificar mis datos.
<b>CONFIRMACIONES:</b>			
~ Se pueden modificar los datos almacenados del proveedor.			
<b>TAREAS:</b>			
~ Diseño de la interfaz de edición del perfil. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-05			

Tabla 18 - Historia de Usuario 17. UH-P-04

UH-P-05			
<b>IDENTIFICADOR:</b>	UH-P-05	<b>TÍTULO:</b>	Añadir Productos
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	3
<b>DEPENDENCIAS:</b>	UH-P-01 y UH-P-02	<b>DESCRIPCIÓN:</b>	Como proveedor, quiero añadir productos para poder venderlos.
<b>CONFIRMACIONES:</b>			
~ Cuando el proveedor registra un producto, todos los datos quedan almacenados en la base de datos. ~ El producto que se pone en venta tiene que pasar por la evaluación del administrador.			
<b>TAREAS:</b>			
~ Diseño de la interfaz de añadir productos. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-17			

Tabla 19 - Historia de Usuario 18. UH-P-05

UH-P-06			
<b>IDENTIFICADOR:</b>	UH-P-06	<b>TÍTULO:</b>	Productos Rechazados
<b>VALOR:</b>	90	<b>ESTIMACIÓN:</b>	3
<b>DEPENDENCIAS:</b>	UH-P-01 y UH-P-02	<b>DESCRIPCIÓN:</b>	Como proveedor, quiero ver los productos que han sido rechazados para poder corregirlos y ponerlos de nuevo a la venta o eliminarlos.
<b>CONFIRMACIONES:</b>			
~ Son productos rechazados cuando han sido denegados por el administrador. ~ Se pueden modificar o eliminar.			
<b>TAREAS:</b>			
~ Diseño de la interfaz de productos rechazados. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-18			

Tabla 20 - Historia de Usuario 19. UH-P-06

UH-P-07			
<b>IDENTIFICADOR:</b>	UH-P-07	<b>TÍTULO:</b>	Editar Producto
<b>VALOR:</b>	80	<b>ESTIMACIÓN:</b>	3
<b>DEPENDENCIAS:</b>	UH-P-06 y UH-P-09	<b>DESCRIPCIÓN:</b>	Como proveedor, quiero acceder a la edición de un producto para poder modificar algún dato.
<b>CONFIRMACIONES:</b>			
~ El proveedor puede modificar de nuevo el producto y actualizarlo. Esta actualización pasará de nuevo a la revisión del administrador.			
<b>TAREAS:</b>			
~ Diseño de la interfaz de edición de productos.			
~ Implementación del código del cliente.			
~ Pruebas de aceptación asociadas			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-19			

Tabla 21 - Historia de Usuario 20. UH-P-07

UH-P-08			
<b>IDENTIFICADOR:</b>	UH-P-08	<b>TÍTULO:</b>	Mis productos
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	13
<b>DEPENDENCIAS:</b>	UH-P-01 y UH-P-02	<b>DESCRIPCIÓN:</b>	Como proveedor, quiero visualizar todos los productos que tengo en venta para poder ver que no tienen ningún fallo.
<b>CONFIRMACIONES:</b>			
~ Se muestran todos los productos que hay en venta (con un límite de 10 por página).			
~ Aparecerá el nombre, el precio actual y el anterior si está en oferta.			
~ Se deben añadir los botones "anterior" y "siguiente" para buscar más productos.			
<b>TAREAS:</b>			
~ Diseño de la interfaz de productos.			
~ Implementación del código del cliente.			
~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-07			

Tabla 22 - Historia de Usuario 21. UH-P-08



UH-P-09			
<b>IDENTIFICADOR:</b>	UH-P-09	<b>TÍTULO:</b>	Visualizar un producto
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	5
<b>DEPENDENCIAS:</b>	UH-P-08	<b>DESCRIPCIÓN:</b>	Como proveedor, quiero visualizar toda la información acerca de un producto para poder comprobar su información.
<b>CONFIRMACIONES:</b>			
~ Se mostrará la imagen del producto. ~ Se mostrará el nombre, la descripción, el precio, la talla y el stock. ~ Se podrá editar, ver los suscriptores que tiene o eliminar.			
<b>TAREAS:</b>			
~ Diseño de la interfaz del producto. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-08			

Tabla 23 - Historia de Usuario 22. UH-P-09

UH-P-10			
<b>IDENTIFICADOR:</b>	UH-P-10	<b>TÍTULO:</b>	Suscriptores
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	5
<b>DEPENDENCIAS:</b>	UH-P-09	<b>DESCRIPCIÓN:</b>	Como proveedor, quiero conocer el tipo de suscriptores que tienen mis productos para poder lograr más ventas añadiendo los productos más deseados.
<b>CONFIRMACIONES:</b>			
~ Información acerca de los suscriptores de un producto.			
<b>TAREAS:</b>			
~ Diseño de la interfaz de suscripción. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-20			

Tabla 24 - Historia de Usuario 24. UH-P-10

UH-P-11			
<b>IDENTIFICADOR:</b>	UH-P-11	<b>TÍTULO:</b>	Facturación
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	5
<b>DEPENDENCIAS:</b>	UH-P-01 y UH-P-02	<b>DESCRIPCIÓN:</b>	Como proveedor, quiero ver los beneficios obtenidos de la venta de mis productos para poder conocer el dinero total que he ganado.
<b>CONFIRMACIONES:</b>			
~ Se puede observar el beneficio total que deja la venta de los productos, así como la información más relevante de cada producto.			
<b>TAREAS:</b>			
~ Diseño de la interfaz de facturación. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-21			

Tabla 25 - Historia de Usuario 25. UH-P-11

UH-P-12			
<b>IDENTIFICADOR:</b>	UH-P-12	<b>TÍTULO:</b>	Contraseña
<b>VALOR:</b>	90	<b>ESTIMACIÓN:</b>	5
<b>DEPENDENCIAS:</b>	-	<b>DESCRIPCIÓN:</b>	Como proveedor, quiero recuperar mi contraseña en caso de olvido para poder iniciar sesión y garantizar la seguridad de los datos.
<b>CONFIRMACIONES:</b>			
~ Se enviará una nueva contraseña al cliente y luego él la podrá modificar.			
<b>TAREAS:</b>			
~ Diseño de la interfaz de recuperación de contraseña. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-09			

Tabla 26 - Historia de Usuario 26. UH-P-12

## Administrador

UH-A-01			
<b>IDENTIFICADOR:</b>	UH-A-01	<b>TÍTULO:</b>	Inicio de sesión
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	2
<b>DEPENDENCIAS:</b>	-	<b>DESCRIPCIÓN:</b>	Como administrador, quiero iniciar o cerrar sesión, para poder acceder o salir del sistema respectivamente.
<b>CONFIRMACIONES:</b>			
~ Al iniciar sesión, se conservarán los datos y se redirigirá a la página principal. ~ Al cerrar sesión, los datos de esa sesión se eliminarán. ~ Al ser administrador, tiene la potestad de modificar sus datos desde la base de datos.			
<b>TAREAS:</b>			
~ Diseño de la interfaz del inicio de sesión. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-01 ~ TEST-02			

Tabla 27 - Historia de Usuario 27. UH-A-01

UH-A-02			
<b>IDENTIFICADOR:</b>	UH-A-02	<b>TÍTULO:</b>	Validar Productos
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	5
<b>DEPENDENCIAS:</b>	UH-A-01	<b>DESCRIPCIÓN:</b>	Como administrador, quiero validar los productos antes de que se pongan en venta para poder comprobar que todos los datos son correctos.
<b>CONFIRMACIONES:</b>			
~ Tiene la potestad de aceptar el producto, si se cumplen los requisitos necesarios. También se puede denegar.			
<b>TAREAS:</b>			
~ Diseño de la interfaz de validar productos. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-22			

Tabla 28 - Historia de Usuario 28. UH-A-02

UH-A-03			
<b>IDENTIFICADOR:</b>	UH-A-03	<b>TÍTULO:</b>	Producto Denegado
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	3
<b>DEPENDENCIAS:</b>	UH-A-02	<b>DESCRIPCIÓN:</b>	Como administrador, quiero denegar un producto si no cumple los requisitos para poder tener una web en condiciones.
<b>CONFIRMACIONES:</b>			
~ Al denegar el producto, el administrador observa todos los datos y la imagen del producto. ~ Tiene un apartado de explicaciones que debe rellenar si lo deniega.			
<b>TAREAS:</b>			
~ Diseño de la interfaz de producto denegado. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-23			

Tabla 29 - Historia de Usuario 29. UH-A-03

UH-A-04			
<b>IDENTIFICADOR:</b>	UH-A-04	<b>TÍTULO:</b>	Mail
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	3
<b>DEPENDENCIAS:</b>	UH-A-03	<b>DESCRIPCIÓN:</b>	Como administrador, quiero enviar un correo al proveedor para poder explicarle de manera detallada la causa del rechazo de un producto.
<b>CONFIRMACIONES:</b>			
~ Se debe crear el mensaje con la explicación del administrador, para que el proveedor pueda modificar de manera cómoda lo que el administrador requiere. ~ La cuenta de proveedor debe ser Gmail.			
<b>TAREAS:</b>			
~ Diseño de la interfaz del mensaje. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-24			

Tabla 30 - Historia de Usuario 30. UH-A-04

UH-A-05			
<b>IDENTIFICADOR:</b>	UH-A-05	<b>TÍTULO:</b>	Ver Clientes/Proveedores
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	5
<b>DEPENDENCIAS:</b>	UH-A-01	<b>DESCRIPCIÓN:</b>	Como administrador, quiero ver que proveedores y usuarios usan mi aplicación para poder tener información acerca de ellos.
<b>CONFIRMACIONES:</b>			
~ Se puede ver toda la información relevante a los clientes y proveedores que están registrados en el sistema.			
<b>TAREAS:</b>			
~ Diseño de la interfaz para ver los clientes y proveedores. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-25			

Tabla 31 - Historia de Usuario 31. UH-A-05

UH-A-06			
<b>IDENTIFICADOR:</b>	UH-A-06	<b>TÍTULO:</b>	Editar Clientes/Proveedores
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	2
<b>DEPENDENCIAS:</b>	UH-A-05	<b>DESCRIPCIÓN:</b>	Como administrador, quiero modificar los datos de los clientes o proveedores si han realizado algo inadecuado o violento para poder tener mi web totalmente controlada y segura.
<b>CONFIRMACIONES:</b>			
~ Se puede modificar todos los campos, incluido el de la contraseña. Así se garantiza que esa persona no use más el sistema.			
<b>TAREAS:</b>			
~ Diseño de la interfaz de edición de los clientes y proveedores. ~ Implementación del código del cliente. ~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-26			

Tabla 32 - Historia de Usuario 32. UH-A-06

UH-A-07			
<b>IDENTIFICADOR:</b>	UH-A-07	<b>TÍTULO:</b>	Conciliación
<b>VALOR:</b>	100	<b>ESTIMACIÓN:</b>	2
<b>DEPENDENCIAS:</b>	UH-A-01	<b>DESCRIPCIÓN:</b>	Como administrador, quiero realizar la conciliación para poder mostrársela a los proveedores.
<b>CONFIRMACIONES:</b>			
~ Se realiza la conciliación de la empresa.			
~ Se realiza la conciliación de los proveedores.			
<b>TAREAS:</b>			
~ Diseño de la interfaz de conciliación.			
~ Implementación del código del cliente.			
~ Pruebas de aceptación asociadas.			
<b>PRUEBAS DE ACEPTACIÓN</b>			
~ TEST-27			

Tabla 33 - Historia de Usuario 33. UH-A-07

## Pruebas

El objetivo de las pruebas de aceptación consiste en verificar el cumplimiento de las historias de usuario que se han definido en el apartado anterior. La información que contiene cada prueba de aceptación realizada es la siguiente:

- ~ **Identificador:** corresponde al identificador exclusivo de cada prueba realizada. Las pruebas de aceptación tendrán un identificador de la forma TEST-XX, donde XX corresponde con un número decimal que sirve para enumerar secuencialmente las pruebas.
- ~ **Título:** nombre para cada prueba.
- ~ **Descripción:** descripción detallada de la prueba.
- ~ **Procedimiento:** detalle de los pasos que se siguen para realizar la prueba.
- ~ **Historia de usuario relacionada:** identifica a la historia de usuario que da origen a la prueba de aceptación.

A continuación, se pueden observar las pruebas de aceptación realizadas para la aplicación. En este caso, no se divide por las partes de la aplicación ya que la mayoría de las pruebas que se realizan son válidas para las diferentes partes.

TEST-01			
<b>IDENTIFICADOR:</b>	TEST-01	<b>TÍTULO:</b>	Inicio de sesión en la aplicación
<b>DESCRIPCIÓN:</b>	Se comprueba que el usuario puede realizar un inicio de sesión en la aplicación.		
<b>PROCEDIMIENTO:</b>	~ El usuario entra en la aplicación. ~ El usuario inserta el email en el campo correspondiente y hace lo propio con la contraseña. ~ El usuario pulsa el botón de "Entrar"		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-U-04, UH-P-02 y UH-A-01		
<b>RESULTADO ESPERADO:</b>	El usuario puede iniciar sesión en el sistema tras introducir los datos correctos de registro.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 34 - Prueba de Aceptación. TEST-01

TEST-02			
<b>IDENTIFICADOR:</b>	TEST-02	<b>TÍTULO:</b>	Cierre de sesión en la aplicación
<b>DESCRIPCIÓN:</b>	Se comprueba que el usuario puede cerrar la sesión en la aplicación.		
<b>PROCEDIMIENTO:</b>	~ El usuario inicia sesión en la aplicación. ~ El usuario accede a la barra de arriba y pulsa "Cerrar Sesión"		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-U-04, UH-P-02 y UH-A-01		
<b>RESULTADO ESPERADO:</b>	El usuario puede cerrar sesión en la aplicación.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 35 - Prueba de Aceptación. TEST-02

TEST-03			
<b>IDENTIFICADOR:</b>	TEST-03	<b>TÍTULO:</b>	Registro en la aplicación
<b>DESCRIPCIÓN:</b>	Se comprueba que el usuario puede registrarse en la aplicación.		
<b>PROCEDIMIENTO:</b>	~ El usuario entra en la aplicación. ~ El usuario accede a la barra de arriba y pulsa "Crear Cuenta". ~ El usuario rellena todos los campos obligatorios y pulsa "Registrarse".		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-U-03 y UH-P-01		
<b>RESULTADO ESPERADO:</b>	El usuario puede registrarse en la aplicación tras introducir correctamente los campos de registro.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 36 - Prueba de Aceptación. TEST-03

TEST-04			
<b>IDENTIFICADOR:</b>	TEST-04	<b>TÍTULO:</b>	Acceder al perfil
<b>DESCRIPCIÓN:</b>	Se comprueba que el usuario puede acceder a su perfil en la aplicación.		
<b>PROCEDIMIENTO:</b>	~ El usuario inicia sesión en la aplicación. ~ El usuario accede a la barra de arriba y pulsa en su nombre.		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-U-11 y UH-P-03		
<b>RESULTADO ESPERADO:</b>	El usuario puede acceder al perfil en la aplicación y visualizar todos sus datos.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 37 - Prueba de Aceptación. TEST-04

TEST-05			
<b>IDENTIFICADOR:</b>	TEST-05	<b>TÍTULO:</b>	Acceder a editar perfil
<b>DESCRIPCIÓN:</b>	Se comprueba que el usuario puede acceder a la edición de su perfil en la aplicación.		
<b>PROCEDIMIENTO:</b>	~ El usuario inicia sesión en la aplicación. ~ El usuario accede a la barra de arriba y pulsa en su nombre. ~ Una vez que el usuario está en el perfil, pulsa "Editar Perfil"		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-U-12 y UH-P-04		
<b>RESULTADO ESPERADO:</b>	El usuario puede modificar sus datos y visualizarlos entrando de nuevo en su perfil.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 38 - Prueba de Aceptación. TEST-05

TEST-06			
<b>IDENTIFICADOR:</b>	TEST-06	<b>TÍTULO:</b>	Visualizar los productos sin iniciar sesión
<b>DESCRIPCIÓN:</b>	Se comprueba que el usuario visualiza los productos sin iniciar sesión en la aplicación.		
<b>PROCEDIMIENTO:</b>	~ El usuario entra en la aplicación. ~ El usuario accede a ver los productos que desea.		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-U-01		
<b>RESULTADO ESPERADO:</b>	El usuario puede ver los productos en la aplicación sin la necesidad de iniciar sesión.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 39 - Prueba de Aceptación. TEST-06



TEST-07			
<b>IDENTIFICADOR:</b>	TEST-07	<b>TÍTULO:</b>	Visualizar los productos con inicio de sesión
<b>DESCRIPCIÓN:</b>	Se comprueba que el usuario visualizar los productos una vez inicia sesión en la aplicación.		
<b>PROCEDIMIENTO:</b>	~ El usuario inicia en la aplicación. ~ El usuario accede a ver los productos que desea.		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-U-01 y UH-P-08		
<b>RESULTADO ESPERADO:</b>	El usuario puede ver los productos en la aplicación después de iniciar sesión.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 40 - Prueba de Aceptación. TEST-07

TEST-08			
<b>IDENTIFICADOR:</b>	TEST-08	<b>TÍTULO:</b>	Acceder a la información de un producto
<b>DESCRIPCIÓN:</b>	Se comprueba que el usuario pueda visualizar toda la información relevante a un producto.		
<b>PROCEDIMIENTO:</b>	~ El usuario entra en la aplicación. ~ Accede a los productos. ~ Pulsa sobre aquel producto del que desea informarse.		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-U-02 y UH-P-09		
<b>RESULTADO ESPERADO:</b>	El usuario puede ver un producto más en detalle pulsando sobre él.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 41 - Prueba de Aceptación. TEST-08

TEST-09			
<b>IDENTIFICADOR:</b>	TEST-09	<b>TÍTULO:</b>	Recuperar Contraseña
<b>DESCRIPCIÓN:</b>	Se comprueba que el usuario pueda cambiar la contraseña en la aplicación.		
<b>PROCEDIMIENTO:</b>	~ El usuario entra en la aplicación. ~ El usuario accede a iniciar sesión y pulsa en "¿Olvidaste tu contraseña?". ~ Introduce su mail y se le enviará una nueva contraseña.		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-U-14 y UH-P-12		
<b>RESULTADO ESPERADO:</b>	El usuario introduce su mail y recibe en el correo su contraseña.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 42 - Prueba de Aceptación. TEST-09

TEST-10			
<b>IDENTIFICADOR:</b>	TEST-10	<b>TÍTULO:</b>	Suscripción a un producto
<b>DESCRIPCIÓN:</b>	Se comprueba que el usuario puede suscribirse a un producto.		
<b>PROCEDIMIENTO:</b>	~ El usuario inicia sesión en la aplicación. ~ Accede a los productos. ~ Pulsa sobre el producto que desee y una vez dentro pulsa el botón "Suscribirse".		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-U-05		
<b>RESULTADO ESPERADO:</b>	El usuario está suscrito correctamente a un producto cuando aparece el botón "Cancelar Suscripción" en él.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 43 - Prueba de Aceptación. TEST-10

TEST-11			
<b>IDENTIFICADOR:</b>	TEST-11	<b>TÍTULO:</b>	Añadir producto al carrito
<b>DESCRIPCIÓN:</b>	Se comprueba que el usuario puede añadir un producto al carrito.		
<b>PROCEDIMIENTO:</b>	~ El usuario inicia sesión en la aplicación. ~ Accede a los productos. ~ Pulsa sobre el producto que desee y una vez dentro pulsa el botón "Comprar".		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-U-06		
<b>RESULTADO ESPERADO:</b>	El usuario puede acceder al carrito y comprobar que se añadió correctamente.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 44 - Prueba de Aceptación. TEST-11

TEST-12			
<b>IDENTIFICADOR:</b>	TEST-12	<b>TÍTULO:</b>	Datos de Compra
<b>DESCRIPCIÓN:</b>	Se comprueba que el usuario puede modificar o validar sus datos de compra.		
<b>PROCEDIMIENTO:</b>	~ El usuario inicia sesión en la aplicación. ~ Accede a los productos. ~ Pulsa sobre el producto que desee y una vez dentro pulsa el botón "Comprar". ~ Una vez que decide no añadir al carrito más productos, pulsa en el botón "Confirmar compra".		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-U-07		
<b>RESULTADO ESPERADO:</b>	El usuario puede validar toda la información a través de la interfaz de la aplicación.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 45 - Prueba de Aceptación. TEST-12

TEST-13			
<b>IDENTIFICADOR:</b>	TEST-13	<b>TÍTULO:</b>	Proceso de Compra
<b>DESCRIPCIÓN:</b>	Se comprueba que el usuario realiza de manera correcta todos los pasos de la compra.		
<b>PROCEDIMIENTO:</b>	<ul style="list-style-type: none"> <li>~ El usuario inicia sesión en la aplicación.</li> <li>~ Accede a los productos.</li> <li>~ Pulsa sobre el producto que desee y una vez dentro pulsa el botón "Comprar".</li> <li>~ Una vez que decide no añadir al carrito más productos, pulsa en el botón "Confirmar compra".</li> <li>~ Revisa sus datos personales, de envío y de pago, y se procede a la compra.</li> </ul>		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-U-08		
<b>RESULTADO ESPERADO:</b>	El usuario realiza el proceso de compra atendiendo a la información que aparece en la interfaz. Lo hace de manera sencilla y clara.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 46 - Prueba de Aceptación. TEST-13

TEST-14			
<b>IDENTIFICADOR:</b>	TEST-14	<b>TÍTULO:</b>	Resumen
<b>DESCRIPCIÓN:</b>	Se comprueba que el usuario visualiza los datos del resumen de compra sin problemas.		
<b>PROCEDIMIENTO:</b>	<ul style="list-style-type: none"> <li>~ El usuario inicia sesión en la aplicación.</li> <li>~ Accede a los productos.</li> <li>~ Pulsa sobre el producto que desee y una vez dentro pulsa el botón "Comprar".</li> <li>~ Una vez que decide no añadir al carrito más productos, pulsa en el botón "Confirmar compra".</li> <li>~ Revisa sus datos personales, de envío y de pago, y se procede a la compra.</li> <li>~ Obtiene la información relativa a la compra (código pedido, vale descuento y fecha de caducidad del vale).</li> </ul>		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-U-09		
<b>RESULTADO ESPERADO:</b>	El usuario visualiza los datos a través de la interfaz de la aplicación.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 47 - Prueba de Aceptación. TEST-14

TEST-15			
<b>IDENTIFICADOR:</b>	TEST-15	<b>TÍTULO:</b>	Mail compra
<b>DESCRIPCIÓN:</b>	Se comprueba que el usuario recibe un email con un pdf al realizar la compra.		
<b>PROCEDIMIENTO:</b>	<ul style="list-style-type: none"> <li>~ El usuario inicia sesión en la aplicación.</li> <li>~ Accede a los productos.</li> <li>~ Pulsa sobre el producto que desee y una vez dentro pulsa el botón "Comprar".</li> <li>~ Una vez que decide no añadir al carrito más productos, pulsa en el botón "Confirmar compra".</li> <li>~ Revisa sus datos personales, de envío y de pago, y se procede a la compra.</li> <li>~ Obtiene la información relativa a la compra (código pedido, vale descuento y fecha de caducidad del vale).</li> <li>~ El usuario debe recibir en su correo de Gmail, un mensaje con un pdf como confirmación de su compra.</li> </ul>		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-U-10		
<b>RESULTADO ESPERADO:</b>	El usuario puede ver en su correo el mail de compra que ha sido enviado.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 48 - Prueba de Aceptación. TEST-15

TEST-16			
<b>IDENTIFICADOR:</b>	TEST-16	<b>TÍTULO:</b>	Acceder Histórico
<b>DESCRIPCIÓN:</b>	Se comprueba que el usuario puede acceder a su histórico de compras.		
<b>PROCEDIMIENTO:</b>	<ul style="list-style-type: none"> <li>~ El usuario inicia sesión en la aplicación.</li> <li>~ Pulsa en la barra de arriba, en su nombre, de esta manera accede a su perfil.</li> <li>~ Una vez dentro sólo tiene que pulsar en el botón "Histórico"</li> </ul>		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-U-14		
<b>RESULTADO ESPERADO:</b>	El usuario observa todas sus compras.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 49 - Prueba de Aceptación. TEST-16

TEST-17			
<b>IDENTIFICADOR:</b>	TEST-17	<b>TÍTULO:</b>	Añadir productos proveedor
<b>DESCRIPCIÓN:</b>	Se comprueba que el proveedor puede añadir sus productos		
<b>PROCEDIMIENTO:</b>	<ul style="list-style-type: none"> <li>~ El proveedor inicia sesión en la aplicación.</li> <li>~ Pulsa en el menú sobre "Añadir Productos".</li> <li>~ Rellena los campos obligatorios y pulsa el botón "Añadir Producto".</li> </ul>		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-P-05		
<b>RESULTADO ESPERADO:</b>	Tras ser aceptados por el administrador, podrá visualizarlos en la zona "Mis Productos".		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 50 - Prueba de Aceptación. TEST-17

TEST-18			
<b>IDENTIFICADOR:</b>	TEST-18	<b>TÍTULO:</b>	Visualizar productos denegados
<b>DESCRIPCIÓN:</b>	Se comprueba que el proveedor puede visualizar los productos denegados.		
<b>PROCEDIMIENTO:</b>	~ El proveedor inicia sesión en la aplicación. ~ Pulsa en el menú sobre "Productos Denegados".		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-P-06		
<b>RESULTADO ESPERADO:</b>	El proveedor puede visualizar los productos que han sido denegados en la zona "Productos Denegados".		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 51 - Prueba de Aceptación. TEST-18

TEST-19			
<b>IDENTIFICADOR:</b>	TEST-19	<b>TÍTULO:</b>	Editar un producto
<b>DESCRIPCIÓN:</b>	Se comprueba que el proveedor puede editar un producto suyo.		
<b>PROCEDIMIENTO:</b>	~ El proveedor inicia sesión en la aplicación. ~ El proveedor puede pulsar en el menú sobre "Productos Denegados" o "Mis Productos". ~ Acceder a un producto y pulsar en el botón de "Editar Producto".		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-P-07		
<b>RESULTADO ESPERADO:</b>	El proveedor visualizará que la edición fue correcta, accediendo a la información de dicho producto.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 52 - Prueba de Aceptación. TEST-19

TEST-20			
<b>IDENTIFICADOR:</b>	TEST-20	<b>TÍTULO:</b>	Visualizar los suscriptores
<b>DESCRIPCIÓN:</b>	Se comprueba que el proveedor puede visualizar los suscriptores de un producto suyo.		
<b>PROCEDIMIENTO:</b>	~ El proveedor inicia sesión en la aplicación. ~ El proveedor pulsa en el menú sobre "Mis Productos". ~ El proveedor accede a un producto y pulsa en el botón de "Suscriptores".		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-P-10		
<b>RESULTADO ESPERADO:</b>	El proveedor visualiza los suscriptores de un producto.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 53 - Prueba de Aceptación. TEST-20

TEST-21			
<b>IDENTIFICADOR:</b>	TEST-21	<b>TÍTULO:</b>	Facturación
<b>DESCRIPCIÓN:</b>	Se comprueba que el proveedor puede visualizar la facturación de la venta de sus productos.		
<b>PROCEDIMIENTO:</b>	~ El proveedor inicia sesión en la aplicación. ~ El proveedor pulsa en el menú sobre "Facturación".		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-P-11		
<b>RESULTADO ESPERADO:</b>	El proveedor visualiza los beneficios que obtiene de la venta de sus productos.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 54 - Prueba de Aceptación. TEST-21

TEST-22			
<b>IDENTIFICADOR:</b>	TEST-22	<b>TÍTULO:</b>	Validación de productos
<b>DESCRIPCIÓN:</b>	Se comprueba que el administrador puede validar correctamente los productos.		
<b>PROCEDIMIENTO:</b>	~ El administrador inicia sesión en la aplicación. ~ El administrador pulsa en el menú sobre "Validar Productos".		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-A-02		
<b>RESULTADO ESPERADO:</b>	El administrador puede validar sin problemas los productos.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 55 - Prueba de Aceptación. TEST-22

TEST-23			
<b>IDENTIFICADOR:</b>	TEST-23	<b>TÍTULO:</b>	Denegación de un producto
<b>DESCRIPCIÓN:</b>	Se comprueba que el proveedor puede visualizar la facturación de la venta de sus productos.		
<b>PROCEDIMIENTO:</b>	~ El administrador inicia sesión en la aplicación. ~ El administrador pulsa en el menú sobre "Validar Productos". Y pulsa en el botón "Denegar" del producto que no cumple los requisitos.		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-A-03		
<b>RESULTADO ESPERADO:</b>	El administrador deniega sin problemas los productos.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 56 - Prueba de Aceptación. TEST-23

TEST-24			
<b>IDENTIFICADOR:</b>	TEST-24	<b>TÍTULO:</b>	Mail denegación producto
<b>DESCRIPCIÓN:</b>	Se comprueba que el proveedor recibe un email con la explicación del administrador.		
<b>PROCEDIMIENTO:</b>	~ El administrador inicia sesión en la aplicación. ~ El administrador pulsa en el menú sobre "Validar Productos". Y a continuación, pulsa en el botón "Denegar" de aquel producto que no cumple los requisitos. ~ Rellena la parte de explicación y pulsa "Enviar".		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-A-04		
<b>RESULTADO ESPERADO:</b>	El proveedor visualiza en su correo el mensaje enviado por el administrador.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 57 - Prueba de Aceptación. TEST-24

TEST-25			
<b>IDENTIFICADOR:</b>	TEST-25	<b>TÍTULO:</b>	Visualización de Clientes/Proveedores
<b>DESCRIPCIÓN:</b>	Se comprueba que el administrador puede visualizar correctamente a los clientes y proveedores.		
<b>PROCEDIMIENTO:</b>	~ El administrador inicia sesión en la aplicación. ~ El administrador pulsa en el menú sobre "Clientes" o "Proveedores".		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-A-05		
<b>RESULTADO ESPERADO:</b>	El administrador visualiza correctamente a sus clientes y proveedores.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 58 - Prueba de Aceptación. TEST-25

TEST-26			
<b>IDENTIFICADOR:</b>	TEST-26	<b>TÍTULO:</b>	Edición de los Clientes/Proveedores
<b>DESCRIPCIÓN:</b>	Se comprueba que el proveedor recibe un email con la explicación del administrador.		
<b>PROCEDIMIENTO:</b>	~ El administrador inicia sesión en la aplicación. ~ El administrador pulsa en el menú sobre "Clientes" o "Proveedores". ~ Tras elegir el cliente/proveedor a editar, pulsa en el botón "Editar". ~ Edita el campo que desea modificar.		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-A-06		
<b>RESULTADO ESPERADO:</b>	El administrador edita correctamente a sus clientes y proveedores.		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 59 - Prueba de Aceptación. TEST-26

TEST-27			
<b>IDENTIFICADOR:</b>	TEST-27	<b>TÍTULO:</b>	Conciliación
<b>DESCRIPCIÓN:</b>	Se comprueba que el administrador realiza la conciliación de la empresa y la de los proveedores.		
<b>PROCEDIMIENTO:</b>	~ El administrador inicia sesión en la aplicación. ~ El administrador pulsa en el menú sobre "Conciliación".		
<b>HISTORIA DE USUARIO RELACIONADA:</b>	UH-A-07		
<b>RESULTADO ESPERADO:</b>	El administrador realiza la conciliación sin problemas		
<b>ESTADO:</b>	Prueba realizada con éxito		

Tabla 60 - Prueba de Aceptación. TEST-27

## Matrices de Trazabilidad entre Historias de Usuarios

### Usuario

	UH-U-01	UH-U-02	UH-U-03	UH-U-04	UH-U-05	UH-U-06	UH-U-07	UH-U-08	UH-U-09	UH-U-10	UH-U-11	UH-U-12	UH-U-13	UH-U-14
UH-U-01														
UH-U-02	X													
UH-U-03				X										
UH-U-04			X											
UH-U-05			X	X										
UH-U-06		X	X	X										
UH-U-07			X	X		X								
UH-U-08			X	X			X							
UH-U-09			X	X				X						
UH-U-10			X	X					X					
UH-U-11			X	X										
UH-U-12											X			
UH-U-13											X			
UH-U-14														

Tabla 61 - Trazabilidad entre Historias de Usuario e Historias de Usuario (Usuario)



## Proveedor

	UH-P-01	UH-P-02	UH-P-03	UH-P-04	UH-P-05	UH-P-06	UH-P-07	UH-P-08	UH-P-09	UH-P-10	UH-P-11	UH-P-12
UH-P-01		X										
UH-P-02	X											
UH-P-03	X	X										
UH-P-04			X									
UH-P-05	X	X										
UH-P-06	X	X										
UH-P-07						X			X			
UH-P-08	X	X										
UH-P-09								X				
UH-P-10									X			
UH-P-11	X	X										
UH-P-12												

Tabla 62 - Trazabilidad entre Historias de Usuario e Historias de Usuario (Proveedor)

## Administrador

	UH-A-01	UH-A-02	UH-A-03	UH-A-04	UH-A-05	UH-A-06	UH-A-07
UH-A-01							
UH-A-02	X						
UH-A-03		X					
UH-A-04			X				
UH-A-05	X						
UH-A-06					X		
UH-A-07	X						

Tabla 63 - Trazabilidad entre Historias de Usuario e Historias de Usuario (Administrador)

## Matriz de Trazabilidad entre Historias de Usuarios y Pruebas

[illegible]

	TEST-15	TEST-16	TEST-17	TEST-18	TEST-19	TEST-20	TEST-21	TEST-22	TEST-23	TEST-24	TEST-25	TEST-26	TEST-26
UH-U-01													
UH-U-02													
UH-U-03													
UH-U-04													
UH-U-05													
UH-U-06													
UH-U-07													
UH-U-08													
UH-U-09													
UH-U-10	X												
UH-U-11													
UH-U-12													
UH-U-13		X											
UH-U-14													
UH-P-01													
UH-P-02													
UH-P-03													
UH-P-04													
UH-P-05			X										
UH-P-06				X									
UH-P-07					X								
UH-P-08													
UH-P-09													
UH-P-10						X							
UH-P-11							X						
UH-P-12													
UH-A-01													
UH-A-02								X					
UH-A-03									X				
UH-A-04										X			
UH-A-05											X		
UH-A-06												X	
UH-A-07													X

Tabla 64 - Matriz de trazabilidad entre Historias de Usuarios y Pruebas

## Capítulo VI – Gestión del Proyecto

A lo largo de este capítulo se expondrá la planificación final que se ha llevado a cabo para desarrollar el sistema y se establecerá el presupuesto del proyecto.

### Planificación

Dado que es uno de los factores más influyentes dentro del desarrollo de cualquier tipo de proyecto o empresa se ha hecho especial énfasis en registrar, planificar y documentar lo que se ha ido elaborando a lo largo de cada *Sprint*. Ya que una buena planificación puede marcar la diferencia entre el éxito y el fracaso y es de vital importancia a la hora de realizar presupuestos y aproximaciones. A continuación se muestra la planificación final de los tres *Sprints* de los que consta el proyecto.

- ~ **Sprint cero o Agile Inception:** se definió el alcance del proyecto, se delimitaron las actividades propuestas y se fomentó la creatividad. Tal y como se expone en *The Ágile Samurai* [40] entre el tutor y yo mismo, definimos un primer esbozo del producto, con una primera aproximación de lo que sería y no sería el proyecto. Nos centramos en el qué y establecimos un *roadmap* para lograr el cómo. Finalizamos dando un esbozo, de la duración estimada (como hemos indicado, se siguió un proceso adaptativo mediante una metodología ágil, ya que no tenemos un plan cerrado de acción, simplemente nos marcamos un horizonte para la realización del proyecto).
- ~ **Sprint 1:** se realizó un análisis de las posibles tecnologías a usar y el tipo de servidor a utilizar, además se implementó las historias de usuario de la siguiente manera:
  - Usuario: UH-U-03, UH-U-04, UH-U-11, UH-U-12 y UH-U-14.
  - Proveedor: UH-P-01, UH-P-02, UH-P-03, UH-P-04 y UH-P-12.
  - Administrador: UH-A-01, UH-A-05 y UH-A-06.
- ~ **Sprint 2:** se implementaron las historias de usuario del siguiente modo:
  - Usuario: UH-U-01, UH-U-02, UH-U-05 y UH-U-06.
  - Proveedor: UH-P-05, UH-P-06, UH-P-07, UH-P-08, UH-P-09 y UH-P-10.
  - Administrador: UH-A-02, UH-A-03 y UH-A-04.
- ~ **Sprint 3:** se implementaron las historias de usuario del siguiente modo:
  - Usuario: UH-U-07, UH-U-08, UH-U-09, UH-U-10 y UH-U-13.
  - Proveedor: UH-P-11.
  - Administrador: UH-A-07.

~ **Sprint 4 cuatro o de cierre:** se realizará una retrospectiva de lo realizado, aprendido y se estudian los errores cometidos a lo largo del desarrollo del proyecto. Este *sprint* se lleva a cabo una vez que ha sido entregado el producto y donde el tutor (*Product Owner*) me encomendará la actividad de realizar una crítica constructiva y un roadmap de evolución del producto.

Dado que el proyecto se ha realizado con una metodología ágil como es *Scrum*, mostrar un diagrama de Gantt no es lo más adecuado, ya que este tipo de diagramas es usado cuando se sigue un plan adaptativo en cascada. La siguiente ilustración muestra a modo aclarativo la planificación del proyecto.

Como en todos los proyectos de este tipo, las fechas pueden estar sujetas a ligeros cambios, no obstante se intentó ajustar al máximo dichas fechas, ya que en caso de tratarse de un proyecto empresarial los retrasos pueden ocasionar pérdidas para la organización.

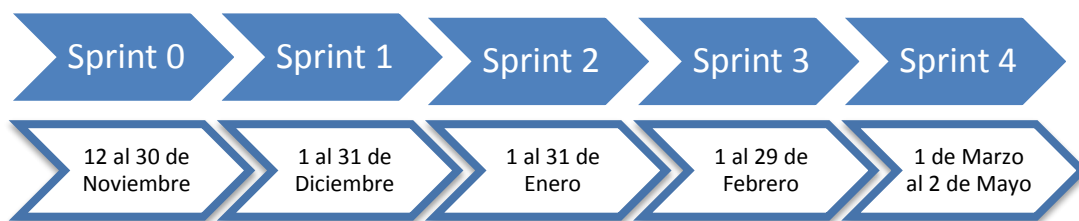


Ilustración 49 – Planificación del proyecto

## Presupuesto del Proyecto

A continuación se parte a elaborar el presupuesto del proyecto. Para ello se tendrá en cuenta el formulario de presupuestos para trabajos de fin de grado y proyectos de fin de carrera de la Universidad Carlos III de Madrid [41].

### Autor

El autor del presente documento es el alumno Roberto Cañizares Sanz.

### Departamento

El presupuesto pertenece al Departamento de Informática de la Universidad Carlos III de Madrid.

## Descripción del Proyecto

El presente proyecto desarrolla un negocio eCommerce de artículos deportivos, donde los usuarios pueden ver todo los productos sin necesidad de registrarse, pero cuando deseen comprar o suscribirse a un producto deben registrarse en la aplicación. Los proveedores, por su parte podrán añadir productos para vender pero tendrán que validarse previamente por el administrador, además los proveedores podrán ver, editar o eliminar sus productos si lo desean. Por su parte el administrador, validará los productos, realizará la conciliación y controlará a usuario y proveedores para que la aplicación funcione correctamente sin ningún tipo de incidente. Esta aplicación se desarrollará completamente en Java y se comunicará con un servidor Glassfish.

## Desglose del Presupuesto

Se va a desglosar en coste del personal y en costes de elementos software y hardware. Finalmente se muestra el coste total del proyecto.

### Coste Personal

Para calcular este coste se van a detallar los gastos asociados al personal, para ello se han tenido en cuenta las siguientes consideraciones.

1. No se contabilizan los siguientes días festivos:
  - ~ 6 y 8 de Diciembre del 2015
  - ~ 1 y 6 de Enero del 2016
  - ~ 24 y 25 de Marzo del 2016
2. El número de días reales de dedicación al proyecto han sido 92 días. Con los siguientes datos acerca de la dedicación se puede obtener el coste asociado al personal al proyecto. Para ello, se va a utilizar la siguiente fórmula:

$$\text{Coste Personal} = \frac{\text{Total Días} * \text{horas/día}}{\text{Dedicación Hombre/mes}} * \text{Coste Hombre/mes}$$

Donde:

- ~ Horas al día de dedicación del equipo de desarrollo = 4
- ~ Total de horas dedicada del *Scrum Manager* y *Product Owner* = 50
- ~ Coste Ingeniero Senior al mes = 4.289,54 €/mes
- ~ Coste Ingeniero Junior al mes = 2.694,39 €/mes
- ~ Dedicación hombre/mes = 131.25 €/mes

NOMBRE	CATEGORÍA	HORAS TOTALES	COSTE TOTAL
ROBERTO CAÑIZARES SANZ	Ingeniero Junior	368	7.554,55 €
JESÚS HERNANDO CORROCHANO	Ingeniero Senior	50	1.634,11 €
TOTAL:			9.188,66 €

Tabla 65 - Coste personal

### Coste Elementos Software y Hardware

El coste del material. En este apartado hace referencia a los diversos equipos informáticos que se preveían necesarios para el desarrollo del trabajo de fin de grado. En la siguiente tabla se mostrará el producto, su precio, el porcentaje de uso dedicado al proyecto, la dedicación (en meses), el periodo de amortización y el coste imputable.

NOMBRE PRODUCTO	PRECIO	% USO DEDICADO AL PROYECTO	DEDICACIÓN (MESES)	PERIODO DE AMORTIZACIÓN	COSTE IMPUTABLE
HP PAVILION 15-P265NS	649 € [42]	100	5	60	64,9 €
TOTAL:					64,9 €

Tabla 66 - Coste elementos software y hardware

Los cálculos realizados en la tabla anterior se rigen a partir de la siguiente fórmula:

$$\text{Cálculo de amortización} = \frac{A}{B} \times C \times D$$

Donde:

- ~ A = número de meses, desde la fecha de facturación, en los que el equipo es utilizado (6 meses).
- ~ B = periodo de amortización (60 meses).
- ~ C = coste del equipo (sin IVA).
- ~ D = % del uso que se dedica al proyecto (100 %).

### Coste Total del Proyecto

Finalmente, se establece el presupuesto total, que integra los costes de los recursos humanos y materiales, costes indirectos y beneficio industrial:

<b>PRECIO TOTAL DEL PROYECTO</b>
<b>COSTE EMPRESARIAL</b>
9.253,56 €
<b>COSTES INDIRECTOS (8 %)</b>
9.993,85 €
<b>MARGEN DE SEGURIDAD (10 %)</b>
10.993,24 €
<b>MARGEN DE BENEFICIO (15 %)</b>
12.642,23 €

Tabla 67 - Precio total del Proyecto

Por lo tanto, el presupuesto del proyecto que describe el presente documento asciende a

**DOCE MIL SEISCIENTOS CUARENTA Y DOS EUROS CON VEINTITRÉS CENTIMOS**

(12.642,23 €) sin IVA.



## Capítulo VII - Conclusiones

Para finalizar el proyecto se expondrán brevemente una serie de conclusiones, la valoración personal del autor y una serie de trabajos futuros cuya finalidad es la de mejorar el acabado del proyecto y apuntar el camino a seguir a partir de este momento.

### Conclusiones y Valoración Personal

Hoy en día, vivimos en un mundo conectado en el que el comercio electrónico es cada vez es más importante. Su uso se incrementa exponencialmente y ya forma parte de la sociedad actual. Gran parte de la sociedad combina el comercio tradicional con el comercio online. Las tiendas que consigan integrar estos dos tipos de negocios son las que mayor ventaja van a obtener sobre las demás.

Uno de los principales motivos que me impulsaron a elegir este proyecto fue el prometedor futuro que tiene por delante y la cantidad inmensa de nuevas tecnologías que se están empezando a usar para potenciar más este tipo de negocio.

La idea principal era realizar un negocio online que tuviera tirón en la actualidad y se decidió realizar un negocio con productos deportivos. ¿Por qué productos deportivos y no otros? Pues bien, actualmente hay un "boom" de gente enganchada a hacer deporte para cuidarse o para sentirse bien con ellos mismos, ya sea por ejemplo, corriendo en la calle o yendo al gimnasio.

Una vez que decidimos el tema de nuestro negocio se realizaron varios análisis, en el primero de ellos se analizó el mercado actual para sacar los puntos fuertes y débiles de aquellas empresas que compiten en nuestro área. En el segundo análisis se seleccionó la arquitectura a emplear y en el último se seleccionó el servidor de aplicaciones más potente y eficaz.

Una vez que tuvimos todo preparado, empezamos las reuniones con el cliente para saber que quería y como lo quería, lo que se denominan "historias de usuario". Tras varias reuniones conseguimos obtener una aplicación web muy eficiente y con un gran potencial, cosa que agradó mucho al cliente.

En este tipo de proyectos se puede sacar tanto conclusiones positivas como negativas. Lo principal es que en un futuro sirvan de ayuda a la hora de elegir el mejor camino a seguir en un nuevo proyecto.

## Conclusiones Positivas

Destacan las siguientes conclusiones positivas:

- ~ **Saber “frenar los pies” a los requisitos propuestos por el cliente.** En este caso concreto, han sido las historias de usuario. El cliente siempre va a querer tener el mejor negocio online, y su mente está en continua lluvia de ideas para lanzar a los responsables del sistema sus pensamientos y características nuevas que añadir y que, por supuesto, quiere que estén listas para la primera versión de despliegue.
- ~ **Adopción de diferentes mentalidades en la realización del proyecto.** La realización de un proyecto de tal calibre, supone todo un reto ya que no se puede pensar solamente en desarrollar todas las funcionalidades que requiere el cliente, sino que ese desarrollo se debe de hacer desde un punto de vista más alejado, como es el del usuario. Se debe desarrollar una aplicación que permita incluir todas las funcionalidades que requiere el cliente y donde los usuarios se sientan cómodos interactuando con ella.
- ~ **Se han perfeccionado conocimientos.** La motivación con la que se empezó el proyecto para afianzar los conocimientos de un grupo de asignaturas de la carrera ha repercutido positivamente. Se han perfeccionado conocimientos en interfaces de usuario, en programación, en bases de datos, en seguridad y en tecnologías informáticas de la web.
- ~ **Cumplir con el objetivo de acabar el trabajo de fin de grado.** Llegados a este punto podemos decir que se han cumplido todos los objetivos que se marcaron al inicio de este apasionante proyecto. Es un momento de satisfacción ver que todo salió mejor de como estaba previsto.
- ~ **Las metodologías o marcos de desarrollo ágil,** son el futuro en el desarrollo de software, al favorecer la comunicación, el compromiso y la sinergia de los equipos. Lo que venimos a denominar Ingeniería Concurrente.

## Conclusiones Negativas

Destacan las siguientes conclusiones negativas:

- ~ **La metodología Scrum, no es apta para equipos de trabajo de dos personas.** Al usar una metodología de desarrollo ágil como ha sido Scrum, se ha podido

comprobar de primera mano cómo esta metodología fue creada para gestionar un equipo de desarrollo de varias personas.

- ~ **Los grandes proyectos no pueden realizarse por una persona.** La realización de un gran proyecto, como ha sido el caso, podría haberse terminado en un plazo menor de tiempo si se hubiese realizado por un conjunto mayor de personas, que estuviesen motivadas y poseyeran los conocimientos necesarios.
- ~ **Velocidad y Quemado.** Ha sido muy difícil para mí calcular los puntos historia que era capaz de quemar un equipo en un *Sprint*, es decir, conocer en cuanto tiempo se va a desarrollar una "historia de usuario" ha sido un trabajo complicado. Una vez acabado el proyecto es cuando podría hacer una estimación más precisa.

## Trabajos Futuros

A continuación se exponen aquellos trabajos opcionales que no han sido posible llevarse a cabo por falta de tiempo y que en caso de comercializar el sistema serían un valor añadido sobre el producto actual que motivaría las ventas. Los trabajos que se tienen pensado como futuros se describen a continuación:

### Interfaz adaptativa

La interfaz del sistema está optimizada para una resolución de una pantalla de ordenador 1920x1080, su objetivo es realizar una interfaz que se adapte sin ningún problema a cualquier dispositivo, permitiendo de ese modo que la sociedad pueda realizar compras de sus productos preferidos desde cualquier lugar y con cualquier dispositivo.

### Redes Sociales

Permitir a la gente que se registre en la aplicación con las diferentes redes sociales como son Facebook, Twitter o Google plus.

### Añadir Comentarios y Valoraciones

Permitir a los usuarios que añadan comentarios y valoraciones sobre los productos, ofreciendo mayor información a los usuarios. También se debe permitir que el proveedor se pueda defender de esos comentarios.

## Añadir Publicidad

La inclusión de publicidad en la aplicación web puede darnos a conocer en muchos sectores, obteniendo esa popularidad que deseamos y además obtenemos un beneficio económico.

## Mejora de algunas funcionalidades

Se podría mejorar el proceso de compra y hacerlo mucho más seguro y confidencial. Cambiar los botones "siguiente" y "anterior" por números de páginas. Y ofrecer más imágenes e información asociada a un producto.

## Recuperar la contraseña

Implementar de manera eficiente, confidencial y segura la recuperación de la contraseña por parte de los clientes.

## Actualizar Stock

Cada vez que se realiza una compra actualizar el stock del producto, esto se puede hacer con un disparador o mediante código java.

## Swagger

El Objetivo de *Swagger* es definir una interfaz estándar y agnóstica del lenguaje (Scala, Java, Javascript, Ruby, PHP o ActionScript) para las API's REST que permita descubrir y comprender las funcionalidades de un servicio sin acceder al código fuente, documentación o mediante la inspección vía web. Cuando definimos apropiadamente un servicio vía *Swagger*, un consumidor puede comprender e interactuar con el servicio remoto con un mínimo de lógica de implementación.

Desde un punto de vista técnico, *Swagger* es una especificación formal rodeada de un gran ecosistema de herramientas, que incluyen todo desde un Front web, librerías de código de bajo nivel y soluciones de gestión para el API de servicios.

Es útil tanto a desarrolladores como a no desarrolladores. Desde el punto de vista de un desarrollador, Swagger se utiliza para describir (definir y documentar) el API de Servicios. En este sentido, nos encontramos con dos aproximaciones:

1. Aproximación "**top-down**": Se utilizaría el Editor de *Swagger* para crear la definición y entonces usar la herramienta *Swagger Codegen* para la genera la implantación.

2. Aproximación "**Bottom-up**": donde se parte de una API REST para la cual se quiere crear la definición *Swagger*. Tanto si se crea la definición manualmente (mediante el uso del Editor de *Swagger*, mencionado anteriormente) como si se une a uno de los frameworks soportados (JX-RS, Node.js, Grails, Scala Play, Spring MVC, Symfony o Ruby) se obtiene la definición *Swagger* del Servicio.

Desde un punto de vista de consumidor del servicio, *Swagger* proporciona una interfaz visual a modo de sandbox dónde podemos testear las llamadas de la API, además de consultar su documentación.

La especificación *Swagger* para la definición y documentación de los servicios se encuentra en la dirección (<http://swagger.io/specification/>)

## Sistema de Logging (Trazas)

El sistema de trazas es una mejora muy importante, por ello, se va a describir el sistema Log4J 2.0, evolución de Log4J. Es el sistema de trazas que más se utiliza en las aplicaciones empresariales, ya que permite al desarrollador controlar qué eventos logar con una granularidad arbitraria: el desarrollador decide qué y dónde se escriben las trazas en cada momento.

Por mi parte, los motivos por los cuales se elegiría Log4J 2.0 son:

- a) Separación de la API. La API de Log4j está separada de la implementación, permitiendo a los desarrolladores de aplicaciones saber qué clases y qué métodos utilizar mientras se asegura la compatibilidad con versiones futuras. Esto permite a los quipos de Log4J mejorar la implementación sin impactar en la API.
- b) Mejoras de rendimiento. Estas mejoras están basadas en la introducción de los *Loggers* asíncronos que en escenarios de máxima concurrencia tienen un *throughput* 18 veces mayor que en la versión Log4J 1.x.
- c) Soporte a múltiples API's incluidas SLF4J and *Commons Logging* APIs.
- d) Recarga automática de las configuraciones.
- e) Filtro avanzado. Se pueden crear filtros para implementar algún tipo de lógica antes de que la acción de paso al Logger. Además los filtros se pueden asociar con los *Loggers*.
- f) Arquitectura basada en *plugins*: *Loggers*, *Appenders* y *LayOut*.
- g) Configurable mediante ficheros properties o xml.
- h) Niveles de Log personalizados.

## Capítulo VIII – Bibliografía

### Bibliografía Referenciada en el Texto

1. Eclipse. *Eclipse for Java Developers*. 2015. [En línea] [Citado el 16 de Enero de 2016]  
<https://eclipse.org/luna/>
2. Google. *Un navegador web rápido y gratuito*. 2015. [En línea] [Citado el 16 de Enero de 2016]  
<https://www.google.es/chrome/browser/desktop/>
3. MySQL. *MySQL Workbench*. 2015. [En línea] [Citado el 16 de Enero de 2016]  
<https://dev.mysql.com/downloads/workbench/>
4. SoapUI. *SoapUI*. 2015. [En línea] [Citado el 16 de Enero de 2016]  
<https://www.soapui.org/>
5. Selenium. *SeleniumHQ Browser Automation*. 2015. [En línea] [Citado el 16 de Enero de 2016]  
<http://www.seleniumhq.org/download/>
6. Insomnia. *Insomnia REST Client*. 2015. [En línea] [Citado el 16 de Enero de 2016]  
<https://chrome.google.com/webstore/detail/insomnia-rest-client/gmodihnfibbjdecbanmpmbmeffnmloel>
7. Alis\_dor. *Historia y evolución de PHP*. Diciembre 2008. [En línea] [Citado el 23 de Enero de 2016]  
<http://es.scribd.com/doc/9150244/Historia-y-Evolucion-de-Php#scribd>
8. Álvarez, Miguel Ángel. *Breve historia de PHP*. Junio 2001. [En línea] [Citado el 30 de Enero de 2016]  
<http://www.desarrolloweb.com/articulos/436.php>
9. Ciberaula. *¿Qué es PHP? Introducción, definición y evolución*. Enero 2013. [En línea] [Citado el 30 de Enero de 2016]  
[http://php.ciberaula.com/articulo/introduccion\\_php/](http://php.ciberaula.com/articulo/introduccion_php/)
10. Berzal Galiano, Fernando; Cortijo Bon, Francisco. *La Plataforma .NET*. Mayo 2013. [En línea] [Citado el 6 de Febrero de 2016]  
<http://elvex.ugr.es/decsai/csharp/dotnet/index.xml>

11. Linares Pellicer, Jordi. *La Plataforma .NET (.NET Framework)* [En línea] [Citado el 6 de Febrero de 2016]  
<http://users.dsic.upv.es/~jlinares/csharp/Tema%201.pdf>
12. Saavedra, Jorge. *¿Qué es Microsoft.NET?* Mayo 2007 [En línea] [Citado el 6 de Febrero de 2016]  
<https://jorgesaavedra.wordpress.com/2007/05/09/%C2%BFque-es-microsoftnet/>
13. Corrochano Hernando, Jesús. *Visión general de JEE*. Septiembre 2014 [En línea] [Citado el 13 de Febrero de 2016]  
[https://aulaglobal.uc3m.es/pluginfile.php/1511192/mod\\_resource/content/1/TEMA%200%20-%20Visi%C3%B3n%20General%20JEE.pdf](https://aulaglobal.uc3m.es/pluginfile.php/1511192/mod_resource/content/1/TEMA%200%20-%20Visi%C3%B3n%20General%20JEE.pdf)
14. Zavala, Daniel. *Introducción a Node.js*. Agosto 2014 [En línea] [Citado el 20 de Febrero de 2016]  
<http://sg.com.mx/content/view/1138>
15. Ray, Óscar. *Una docena de conceptos que deberías conocer sobre Node.js*. Noviembre 2014 [En línea] [Citado el 15 de Octubre de 2015]  
<http://unadocenade.com/una-docena-de-conceptos-que-deberias-conocer-node-js/>
16. Corrochano Hernando, Jesús. *EJB*. Noviembre 2014 [En línea] [Citado el 21 de Febrero de 2016]  
[https://aulaglobal.uc3m.es/pluginfile.php/1511195/mod\\_resource/content/1/TEMA%205-EJBs.pdf](https://aulaglobal.uc3m.es/pluginfile.php/1511195/mod_resource/content/1/TEMA%205-EJBs.pdf)
17. Rodríguez, Anderson. *Historia de MySQL*. Marzo 2011 [En línea] [Citado el 27 de Febrero de 2016]  
[http://es.slideshare.net/anderson\\_rodriguez/historia-de-mysql](http://es.slideshare.net/anderson_rodriguez/historia-de-mysql)
18. Enríquez Toledo, Alma; Maldonado Ayala, Jesús; Nakamura Ortega, Yunko; Nogueron Toledo, Goretty. *MySQL*. Septiembre 2013 [En línea] [Citado el 27 de Febrero de 2016]  
<http://www.gridmorelos.uaem.mx/~mcruz//cursos/miic/MySQL.pdf>
19. Nguba, Rolando. *Computación en la nube*. Marzo de 2014 [En línea] [Citado el 28 de Febrero de 2016]  
<https://aula128.wordpress.com/2014/03/29/computacion-en-la-nube-cloud-computing/>
20. Barranco Fragoso, Ricardo. *¿Qué es Big Data?* Mayo de 2012 [En línea] [Citado el 5 de Marzo de 2016]  
<https://www.ibm.com/developerworks/ssa/local/im/que-es-big-data/>

21. Puppet Labs. *¿Qué es Puppet?* [En línea] [Citado el 5 de Marzo de 2016]  
<https://puppetlabs.com/puppet/what-is-puppet>
22. Velasco, Rubén. *Docker, qué es y cómo funciona la virtualización de contenedores*. Febrero de 2016. [En línea] [Citado el 6 de Marzo de 2016]  
<http://www.redeszone.net/2016/02/24/docker-funciona-la-virtualizacion-contenedores/>
23. García Oterino, Ana María del Carmen. *¿Qué es Docker?* Julio del 2015 [En línea] [Citado el 6 de Marzo de 2016]  
<http://www.javiergarzas.com/2015/07/que-es-docker-sencillo.html>
24. HostDime Blog. *Linux para una red Cloud y centro de datos con CoreOs*. 2015 [En línea] [Citado el 12 de Marzo de 2016]  
<http://blog.hostdime.com.co/linux-para-una-red-cloud-y-centro-de-datos-con-coreos/>
25. García Oterino, Ana María del Carmen. *¿Qué es eso de los Microservicios?* Junio del 2015 [En línea] [Citado el 12 de Marzo de 2016]  
<http://www.javiergarzas.com/2015/06/microservicios.html>
26. Álvarez Caules, Cecilio. *¿Qué es un Microservicio?* Enero de 2015 [En línea] [Citado el 12 de Marzo de 2016]  
<http://www.arquitecturajava.com/que-es-un-microservicio/>
27. Romero, Hermenegildo. *Metodologías de Desarrollo*. Febrero de 2012 [En línea] [Citado el 19 de Marzo de 2016]  
<http://es.slideshare.net/MeneRomero/metodologias-de-desarrollo>
28. Royce, W. (1970). *Managing the Development of Large Software Systems*. 2013 [En línea] [Citado el 19 de Marzo de 2016]  
<http://www.cs.umd.edu/class/spring2003/cmsc838p/Process/waterfall.pdf>
29. Cataldi, Z., Lage, F., Pessacq, R., García Martínez, R. *Ingeniería de Software Educativo*. [En línea] [Citado el 19 de Marzo de 2016]  
<http://www.iidia.com.ar/rgm/comunicaciones/c-icie99-ingenieriasoftwareeducativo.pdf>
30. Canós H., José; Letelier, Patricio; Penadés, M<sup>a</sup> Carmen. *Metodologías Ágiles en el Desarrollo de Software*. [En línea] [Citado el 20 de Marzo de 2016]  
[http://noqualityinside.com/nqi/nqifiles/XP\\_Agil.pdf](http://noqualityinside.com/nqi/nqifiles/XP_Agil.pdf)
31. Raya, Raúl. *¿Qué son las metodologías ágiles?* Julio 2014 [En línea] [Citado el 20 de Marzo de 2016]  
<http://blog.leanmonitor.com/es/que-son-las-metodologias-agiles/>



32. VersionOne Inc. *9<sup>TH</sup> State of Agile Survey*. 2015. [En línea] [Citado el 9 de Abril de 2016]  
<https://www.versionone.com/pdf/state-of-agile-development-survey-ninth.pdf>
33. Garzás, J. *Gestión de proyectos ágil... y las experiencias de más de 12 años de proyectos ágiles*. 2014. 233gradosTI.
34. Schwaber, H. Sutherland, J. *The Scrum Guide – The Definitive Guide to Scrum: The Rules of the Game*. Julio 2013. [En línea] [Citado el 9 de Abril de 2016]  
<http://www.scrumguides.org/docs/scrumguide/v1/scrum-guide-us.pdf>
35. Trello. *Trello Web Application*. 2014. [En línea] [Citado el 10 de Abril de 2016]  
<https://trello.com/>
36. Skype. *Free calls to friend and family*. 2014. [En línea] [Citado el 10 de Abril de 2016]  
<http://www.skype.com/es/>
37. Dropbox Inc. *Tus cosas, siempre contigo*. 2015. [En línea] [Citado el 10 de Abril de 2016]  
<https://www.dropbox.com/home>
38. Jeffries, R. 2001. *Essential XP: Card, conversation, confirmation*. [En línea] [Citado el 9 de Abril de 2016]  
<http://xprogramming.com/articles/expcardconversationconfirmation>
39. Cohn, M. (2005). *Agile estimating and planning*. Upper Saddle River, NJ, USA. Prentice Hall PTR.
40. Ryan Singer. *The Agile Warrior*. Noviembre de 2010. [En línea] [Citado el 9 de Abril de 2016]  
<https://agilewarrior.wordpress.com/2010/11/06/the-agile-inception-deck/>
41. Universidad Carlos III de Madrid. *Plantilla para presupuestos de Proyectos de Fin de Carrera y Trabajos de Fin de Grado*. 2010. [En línea] [Citado el 16 de Abril de 2016]  
[http://www.uc3m.es/portal/page/portal/administracion\\_campus\\_leganes\\_est\\_cg/proyecto\\_fin\\_carrera/Formulario\\_PresupuestoPFC-TFG%20\(3\)\\_1.xlsx](http://www.uc3m.es/portal/page/portal/administracion_campus_leganes_est_cg/proyecto_fin_carrera/Formulario_PresupuestoPFC-TFG%20(3)_1.xlsx)
42. *Portátil HP Pavilion*. [En línea] [Citado el 16 de Abril de 2016]  
<https://www.elcorteingles.es/electronica/A16251614-portatil-hp-pavilion-156-15-ab102ns-amd-quad-core-a10-8700p/>

## Capítulo IX – Glosario de Términos

A continuación se incluyen todos aquellos términos poco conocidos o de difícil interpretación:

- ~ **CGI**, tecnología de la *World Wide Web* que permite a un cliente solicitar datos de un programa ejecutado en un servidor web.
- ~ **Clave Foránea**, identifica una columna o grupo de columnas en una tabla (tabla referenciante) que se refiere a una columna o grupo de columnas en otra tabla (tabla referenciada) en bases de datos relacionales.
- ~ **eCommerce** (del inglés *electronic commerce*), consiste en la distribución, marketing, venta, compra y suministro de información de productos o servicios a través de Internet.
- ~ **Framework**, marco de aplicación o conjunto de bibliotecas orientadas a la reutilización de componentes software para el desarrollo de aplicaciones.
- ~ **Hypervisor** (del inglés *Hypervisor*), es una plataforma que permite aplicar diversas técnicas de control de virtualización para utilizar, al mismo tiempo, diferentes sistemas operativos en un mismo ordenador.
- ~ **HTML** (del inglés *HyperText Markup Language*), lenguaje de programación utilizado para el desarrollo de páginas web.
- ~ **Inception** es un conjunto de dinámicas orientadas a enfocar a todas las personas involucradas en un proyecto hacia un mismo objetivo, reduciendo muchas de las incertidumbres, ayudando a explicitar los riesgos más evidentes y poniendo en común las expectativas de todos.
- ~ **J2EE** (*Java 2 Enterprise Edition*), es una arquitectura multicapa para implementar aplicaciones de tipo empresarial y aplicaciones basadas en la Web.
- ~ **Java**, lenguaje de programación orientado a objetos.
- ~ **Javascript**, lenguaje de programación web orientado a objetos.
- ~ **JMS** (del inglés *Java Message Service*), estándar que especifica cómo las aplicaciones de Java envían y reciben mensajes.
- ~ **JPA** (del inglés *Java Persistence API*), proporciona un modelo de persistencia para mapear bases de datos relacionales en Java.
- ~ **Metodología de desarrollo Lean** es un modelo de gestión que propone maximizar el valor para los clientes, reduciendo los costes y aumentando la calidad del producto o del servicio.

- ~ **REST** (del inglés *REpresentational State Transfer*), es un estilo arquitectónico que está basado en el concepto de transferencia entre sistemas a través de la red.
- ~ **RoadMap** (se podría traducir como hoja de ruta) es una planificación del desarrollo software con los objetivos a corto y largo plazo.
- ~ **RPC** (del inglés *Remote Procedure Call*), es un protocolo de red que permite a un programa de ordenador ejecutar código en otra máquina remota sin tener que preocuparse por las comunicaciones entre ambas.
- ~ **Sistema Operativo**, programa que gestiona los recursos hardware de un sistema informático.
- ~ **UNIX**, sistema operativo portable, multitarea y multiusuario.

## Anexo A – Manual de Usuario

En este anexo, se hará una revisión de la interfaz de usuario para ayudar ante las posibles dudas que puedan surgir con el uso de la aplicación, tanto para usuario como para proveedores.

### Usuario

A continuación se detallan las acciones que puede realizar.

#### Registro de sesión

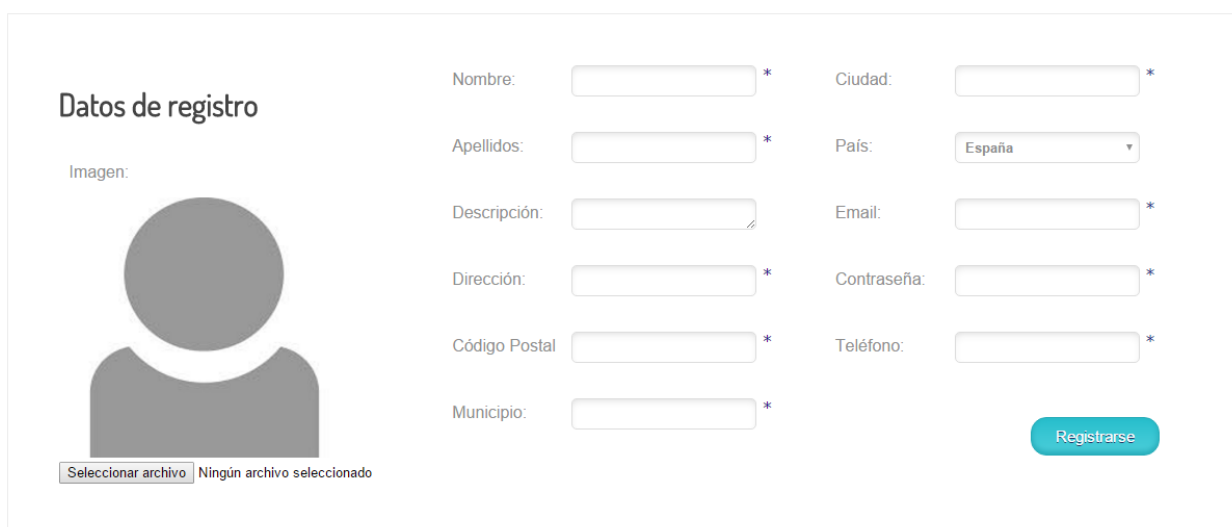
Antes de registrar un nuevo usuario en la aplicación, se tiene que tener cerrada la sesión del usuario actual. Ver sección “Cierre de sesión”.

Si se tiene cerrada la sesión, o es la primera vez que el usuario accede a la aplicación y desea registrarse, podrá pulsar el botón “Crear cuenta” que encontrará en la parte superior de la página.

**Iniciar Sesión / Crear Cuenta / Conócenos**

Ilustración 50 – Visualización botón Crear cuenta Usuario

En la nueva ventana desplegada, se deberá introducir nombre y apellidos, una descripción, una dirección para enviar los pedidos, un código postal, un municipio, una ciudad, un país, un email, una contraseña, un teléfono de contacto y una imagen de perfil. Una vez introducidos los datos, se debe pulsar el botón “Registrarse” para crear la nueva cuenta de usuario.



El formulario de registro de usuario se divide en dos secciones principales. A la izquierda, titulada "Datos de registro", hay un campo para la "Imagen:" con un icono de perfil gris y un botón "Seleccionar archivo" que indica "Ningún archivo seleccionado". A la derecha, hay una serie de campos de entrada con etiquetas y asteriscos obligatorios:

- Nombre:  \*
- Apellidos:  \*
- Descripción:  \*
- Dirección:  \*
- Código Postal:  \*
- Municipio:  \*
- Ciudad:  \*
- País:  ▼
- Email:  \*
- Contraseña:  \*
- Teléfono:  \*

En la esquina inferior derecha del formulario hay un botón azul con el texto "Registrarse".

Ilustración 51 - Creación de un nuevo usuario

## Inicio de sesión

Antes de iniciar sesión con un usuario en el sistema, se tiene que tener cerrada la sesión de otro usuario. Ver sección "Cierre de sesión".

Si se tiene cerrada la sesión, o es la primera vez que el usuario accede a la aplicación web, éste podrá navegar sin problemas sobre ella. Pero no podrá realizar acciones como suscribirse a un producto o comprarlo. Para eso, antes debe iniciar sesión.

En la pantalla desplegada, se debe introducir el email y la contraseña. Una vez introducidos los datos se debe pulsar el botón "Entrar" y se producirá un cambio en la parte de superior de la página.

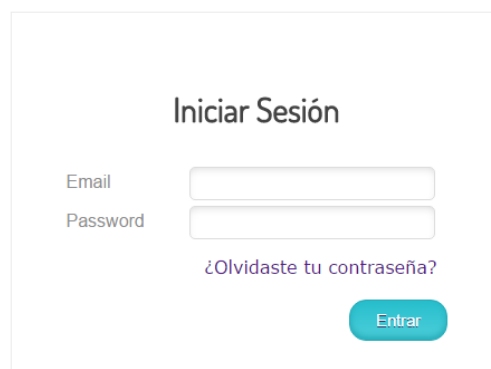
El formulario de inicio de sesión tiene un título "Iniciar Sesión" en el centro. Debajo del título hay dos campos de entrada: "Email" y "Password". Debajo de estos campos hay un enlace que dice "¿Olvidaste tu contraseña?". En la parte inferior derecha del formulario hay un botón verde con el texto "Entrar".

Ilustración 52 - Inicio de Sesión Usuario



Ilustración 53 - Cambio en la interfaz al iniciar de sesión en Usuario

## Cierre de sesión

Acceder a la parte de arriba de la interfaz y pulsar sobre el botón "Cerrar sesión", una vez pulsado el usuario habrá abandonado la sesión.




Ilustración 54 - Cambio en la interfaz al cerrar sesión en Usuario

## Acceder al perfil

Para que el usuario pueda ver su perfil, sólo deberá acceder a la parte superior de la interfaz y pulsar sobre su *nombre*. Una vez pulsado, se desplegará una nueva página en la que podrá observar su perfil y podrá editarlo o ver su historial.

### Mi Perfil



Nombre:	Roberto	País:	Spain
Apellidos:	Canizares	Teléfono:	654782139
Email:	rober93c@gmail.com	Descripción:	Usuario muy activo del running
Dirección:	C/Gainza		
Código Postal	28041		
Municipio:	Madrid		
Ciudad:	Madrid		

[Editar Perfil](#)
[Historico](#)


Ilustración 55 - Perfil Usuario

## Editar el perfil

Para acceder a la edición del perfil, el usuario deberá acceder primero a su perfil (Ver sección "Acceder al perfil"), una vez dentro deberá pulsar el botón "Editar Perfil". Se desplegará una nueva página con los datos del usuario. Se podrán cambiar todos aquellos datos que se desean. Una vez que se concluyen los cambios se deberán guardar de nuevo en el sistema, para ello se deberá pulsar el botón "Guardar Cambios".

### Editar Perfil

Imagen:



Ningún archivo seleccionado

Nombre:	<input type="text" value="Roberto"/> *	Ciudad:	<input type="text" value="Madrid"/> *
Apellidos:	<input type="text" value="Canizares"/> *	País:	<input type="text" value="España"/> ▼
Descripción:	<input type="text" value="Usuario muy activo del running"/> ▼	Email:	<input type="text" value="rober93c@gmail.com"/> *
Dirección:	<input type="text" value="C/Gainza"/> *	Contraseña:	<input type="password" value="*****"/> *
Código Postal	<input type="text" value="28041"/> *	Teléfono:	<input type="text" value="654782139"/> *
Municipio:	<input type="text" value="Madrid"/> *		

Ilustración 56 - Editar Perfil Usuario

## Histórico

Para acceder al histórico de compras del usuario, el usuario deberá acceder primero a su perfil (Ver sección “Acceder al perfil”), una vez dentro, deberá pulsar el botón “Histórico”.

Se desplegará una nueva página en la que se podrá ver los productos comprados, su proveedor, el precio que tenían y el total de dinero gastado en todas las compras.

Imagen	Producto	Proveedor	Precio	TOTAL
	Gafas	otrebora430@gmail.com	24,50 €	1162,59 €
	Reloj Tom Tom	otrebora430@gmail.com	148,50 €	
	Asisc Amarillas	otrebora430@gmail.com	84,90 €	
<div>Anterior</div> <div>Siguiente</div>				

Ilustración 57 - Histórico

## Acceder a productos

El usuario puede acceder a los productos sin la necesidad de iniciar sesión en la aplicación. Pero si desea primero iniciar sesión (Ver sección “Inicio de sesión”) y luego ver los productos también se puede. Basta con acceder al menú y ver aquellos productos que el usuario desee.

Hombre	Mujer	Niños	Accesorios	Suscripciones
--------	-------	-------	------------	---------------

Ilustración 58 - Menú productos

Una vez que pulsa en cualquiera de los botones que se observan en la ilustración de arriba, se desplegará una nueva página mostrando dichos productos.

<div>Por Nombre ▼</div> <div>Mostrando 2 de 2 Producto(s)</div>	
 <p>Camiseta tirantes</p> <p>9,50 €</p>	 <p>Pantalon adidas</p> <p>15,99 € <small>Antes 20,99 €</small></p>
<div>Anterior</div> <div>Siguiente</div>	

Ilustración 59 - Ejemplo productos mujer

## Acceder al producto

Para ver toda la información acerca de un producto, basta con ir a los productos que desea (Ver sección "Acceder a productos") y pulsar encima de uno. La nueva página que se desplegará nos ofrecerá toda la información acerca del producto.

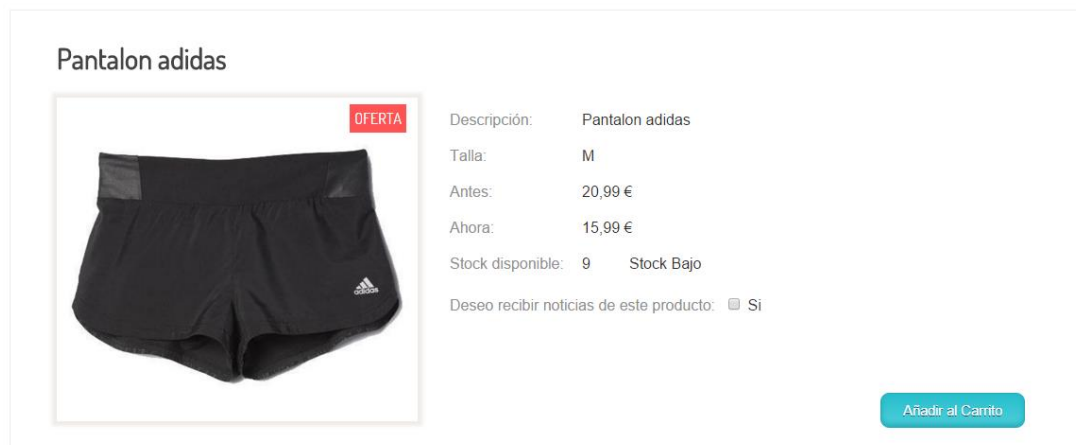


Ilustración 60 - Información acerca de un producto

## Añadir a suscripciones

Para añadir un producto a tus suscripciones, lo único que debes hacer es acceder al producto (Ver sección "Acceder al producto") y pulsar en "Deseo recibir noticias de este producto" y pulsar el botón "Suscribirse". Una vez realizó estos pasos, saldrá un botón que pone "Cancelar Suscripción" por si lo deseas.

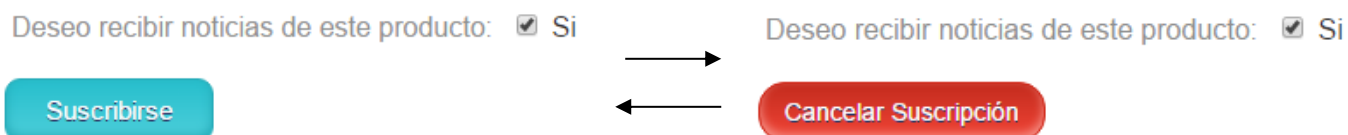


Ilustración 61 - Añadir/Cancelar Suscripción

## Acceder a suscripciones

Si un usuario desea ver sus suscripciones, deberá acceder al menú y pulsar sobre el botón "Suscripciones". La página desplegada le mostrará la imagen del producto suscrito, una descripción suya, su precio y el proveedor al que pertenece. Además podrá cancelar la suscripción si lo desea, nada más que pulsando el botón "Cancelar Suscripción".



Nombre	Descripción	Precio	Proveedor	Imagen	Cancelar Suscripción
Gafas	Gafas de running para protegerte del sol cada vez que salgas a hacer deporte.	24,50 €	otrebor430@gmail.com		Cancelar Suscripción
Guantes	Guantes Adidas para protegerte del frío en estos meses de invierno.	8,50 €	otrebor430@gmail.com		Cancelar Suscripción
Reloj Polar	Reloj con GPS, pulsometro y sumergible a 50 metros.	124,50 €	otrebor430@gmail.com		Cancelar Suscripción
				Anterior	Siguiente

Ilustración 62 - Suscripciones de productos

## Añadir/Acceder al carrito

Si un usuario desea añadir un producto al carrito deberá acceder a él (Ver sección “Acceder al producto”) y pulsar sobre el botón “Añadir al carrito”.

La página desplegada será la misma que si pulsamos en el menú de la parte superior en el botón “Carrito”. En esta página se podrá contemplar el producto, su descripción y su precio.

Producto	Precio	Eliminar Producto	PEDIDO
	Pantalon adidas Pantalon adidas 15,99€	Eliminar producto	Subtotal antes del coste de envio 15,99€ Envio (3-5 dias) 0,00€ Total 15,99€ Confirmar compra

Ilustración 63 - Carrito Usuario

## Realizar la compra

Para acceder al proceso de compra, el usuario debe añadir los productos en el carrito (Ver sección “Añadir/Acceder al carrito”) y una vez terminada la compra deberá pulsar en el botón “Confirma Compra”.

La página desplegada mostrará los datos personales y de envío del usuario, los cuales deberá aceptar para seguir con la compra.

The screenshot displays a registration form with two main sections: 'Datos Personales' (Personal Data) and 'Datos del Envío' (Shipping Data). The 'Datos Personales' section includes fields for Name (Roberto), Surnames (Canizares), Address (C/Gainza), Email (rober93c@gmail.com), and Phone (654782139). The 'Datos del Envío' section is identical but currently empty. A checkbox '¿Usar estos datos?' is checked, with 'Si' selected. At the bottom right, there are 'Atrás' and 'Continuar' buttons.

Ilustración 64 – Visualización de los datos de personales y envío

Tras pulsar en el botón “Continuar”, se desplegará una página donde se deberá proceder a introducir los datos para finalizar la compra.

Tras introducir los datos que se piden, como un número de tarjeta (16 dígitos), una fecha de vencimiento y el código de seguridad de la tarjeta. Se podrá pulsar el botón “Comprar Ahora”.

### Forma de Pago

Forma de Pago: MasterCard \*

Número de tarjeta:

Fecha de Vencimiento: 09/16

Código de Seguridad:

### Introduzca vale de descuento

Código

**Precio Total: 15,99 €**

Ilustración 65 - Forma de Pago

Una vez que lo pulsemos se desplegará una página resumen, para confirmar los datos y proceder a la compra de una manera segura.

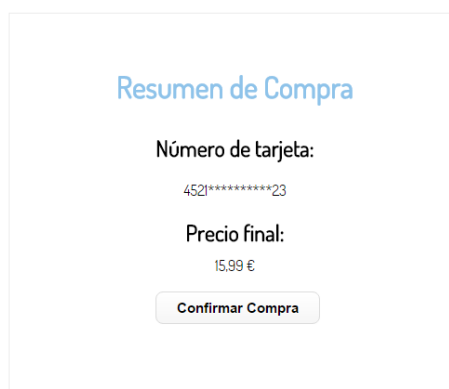


Ilustración 66 - Confirmación de la compra

## Proveedor

A continuación se detallan las acciones que puede realizar.

### Registro de sesión

Antes de registrar un nuevo proveedor en la aplicación, se tiene que tener cerrada la sesión del proveedor actual. Ver sección "Cierre de sesión".

Si se tiene cerrada la sesión, o es la primera vez que el proveedor accede a la aplicación y desea registrarse, podrá pulsar el botón "Crear cuenta" que encontrará en la parte superior de la página.

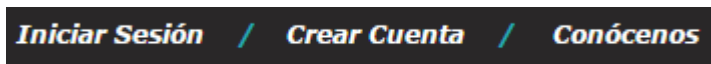


Ilustración 67 - Visualización botón Crear cuenta Proveedor

En la nueva ventana desplegada, se deberá introducir nombre y apellidos, una descripción, una dirección, un código postal, un municipio, una ciudad, un país, un email, una contraseña, un teléfono de contacto y una imagen de perfil. Una vez introducidos los datos, se debe pulsar el botón "Registrarse" para crear la nueva cuenta de proveedor.

A screenshot of a registration form titled 'Datos de registro'. On the left, there is a profile picture placeholder with a 'Seleccionar archivo' button and the text 'Ningún archivo seleccionado'. The form contains the following fields: 'Nombre' (required), 'Apellidos' (required), 'Descripción', 'Dirección' (required), 'Código Postal' (required), 'Municipio' (required), 'Ciudad' (required), 'País' (dropdown menu with 'España' selected), 'Email' (required), 'Contraseña' (required), and 'Teléfono' (required). A 'Registrarse' button is located at the bottom right.

Ilustración 68 - Creación de un nuevo usuario

## Inicio de sesión

Antes de iniciar sesión con un proveedor en el sistema, se tiene que tener cerrada la sesión de otro proveedor. Ver sección "Cierre de sesión".

Si se tiene cerrada la sesión, o es la primera vez que el proveedor accede a la aplicación web, deberá iniciar sesión para poder hacer uso de todas las funcionalidades.

En la pantalla desplegada, se debe introducir el email y la contraseña. Una vez introducidos los datos se debe pulsar el botón "Entrar" y se producirá un cambio en la parte de superior de la página.

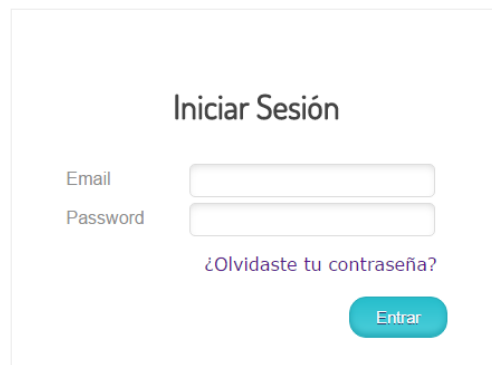
El formulario de inicio de sesión tiene un título "Iniciar Sesión" en el centro. Debajo del título hay dos campos de entrada: "Email" y "Password". Debajo de estos campos hay un enlace que dice "¿Olvidaste tu contraseña?". En la parte inferior derecha del formulario hay un botón azul con el texto "Entrar".

Ilustración 69 - Inicio de Sesión Proveedor



Ilustración 70 - Cambio en la interfaz al iniciar de sesión del Proveedor

## Cierre de sesión

Acceder a la parte de arriba de la interfaz y pulsar sobre el botón "Cerrar sesión", una vez pulsado el usuario habrá abandonado la sesión.

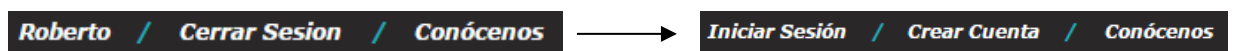



Ilustración 71 - Cambio en la interfaz al cerrar sesión en Proveedor

## Acceder al perfil

Para que el proveedor pueda ver su perfil, sólo deberá acceder a la parte de arriba de la interfaz y pulsar sobre su *nombre*. Una vez pulsado, se desplegará una nueva página en la que podrá observar su perfil y podrá editarlo o ver el historial.

### Mi Perfil



Nombre:	Roberto	Ciudad:	Madrid
Apellidos:	Canizares Sanz	País:	Spain
Email:	otrebtor430@gmail.com	Teléfono:	666548787
Dirección:	C/Gainza	Descripción:	Proveedor muy eficiente
Código Postal	28041		
Municipio:	Madrid		

Editar Perfil


Ilustración 72 - Perfil Proveedor

## Editar el perfil

Para acceder a la edición del perfil, el proveedor deberá acceder, primero a su perfil (Ver sección “Acceder al perfil”) y una vez dentro, deberá pulsar el botón “Editar Perfil”. Se desplegará una nueva página con los datos del usuario, se podrán cambiar todos aquellos datos que se desean. Una vez que los cambios han concluido se deberán guardar de nuevo en el sistema, para ello se deberá pulsar el botón “Guardar Cambios”.

### Editar Perfil

Imagen:



Seleccionar archivo Ningún archivo seleccionado

Nombre:	<input type="text" value="Roberto"/>	*	Ciudad:	<input type="text" value="Madrid"/>	*
Apellidos:	<input type="text" value="Canizares Sanz"/>	*	País:	<input type="text" value="España"/>	▼
Descripción:	<input type="text" value="Proveedor muy eficiente"/>		Email:	<input type="text" value="otrebtor430@gmail.com"/>	*
Dirección:	<input type="text" value="C/Gainza"/>	*	Contraseña:	<input type="password" value="*****"/>	*
Código Postal	<input type="text" value="28041"/>	*	Teléfono:	<input type="text" value="666548787"/>	*
Municipio:	<input type="text" value="Madrid"/>	*			

Guardar Cambios

Ilustración 73 - Editar Perfil Proveedor


## Añadir a productos

Cuando el proveedor desee poner en venta sus productos, este deberá acceder al menú y pulsar el botón "Añadir Productos".

Una vez desplegada la nueva página, deberá poner una imagen del producto, un nombre, una descripción, un número de unidades, un tipo (zapatilla, ropa o accesorio) y una talla, además de un precio de venta con un mínimo y un máximo para que después el administrador del sistema decida a qué precio se tiene que ajustar el producto. También puede indicar si un producto está en oferta o no.

**Datos del Producto**

Imagen Producto:



Ningún archivo seleccionado

Nombre del Producto:  \*

Descripción del producto:  \*

Número de unidades:  \*

Tipo:  \*

Precio de Venta:  \*

Precio mínimo de venta:  \*

Precio máximo de venta:  \*

¿Está de oferta?: ☐ Si

Ilustración 74- Añadir Producto Proveedor

## Productos denegados

El proveedor puede acceder a sus productos denegados pulsando sobre el botón del menú "Productos Denegados". Una vez dentro, podrá editarlos o eliminarlos si no desea ponerlos en venta. Si estos productos son denegados por algún motivo, el administrador lo indica a través del mensaje al proveedor.


Imagen	Nombre	Descripción	Unidades	Tipo	Talla	Precio	Oferta	Editar	Eliminar
	Asics Rojas	Asics Rojas	9	zapatillaM	38	84,50€	No	<input type="button" value="Editar"/>	<input type="button" value="Eliminar"/>
									<input type="button" value="Anterior"/> <input type="button" value="Siguiente"/>

Ilustración 75 - Productos denegados

## Acceder a productos

El proveedor solo podrá ver los productos que tiene en venta, para ello deberá acceder al menú y pulsar en el botón “Mis productos”. Se desplegará una página nueva mostrando todos sus productos a la venta.

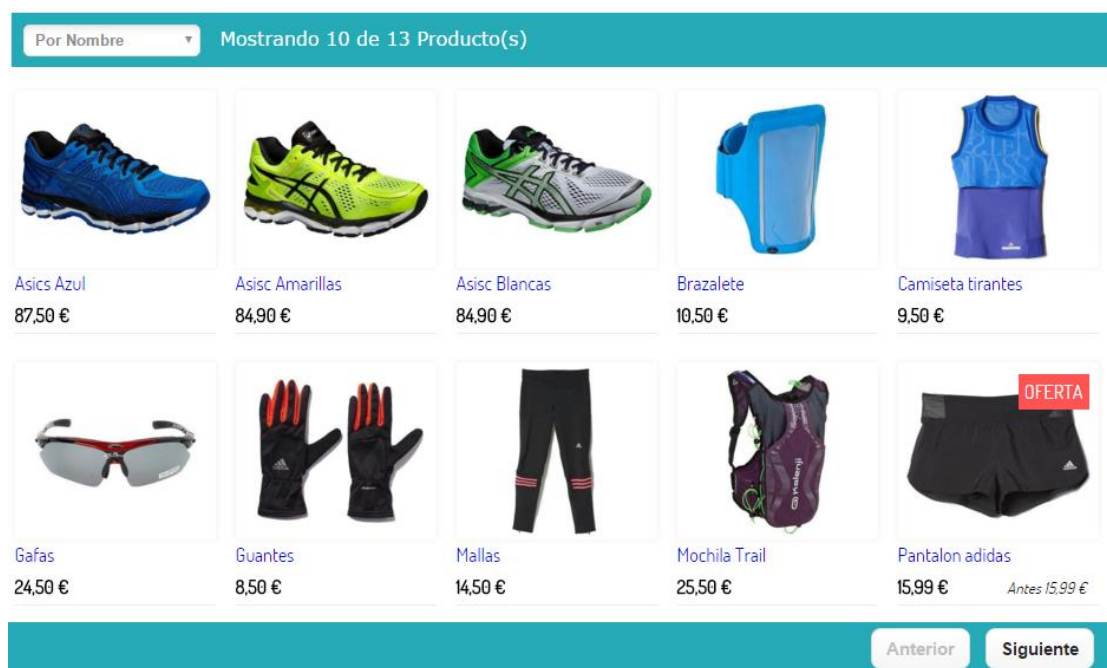


Ilustración 76 - Productos Proveedor

## Acceder al producto

Para acceder a un producto, debe pasar por la sección “Acceder Productos”. Una vez que vea todos los productos sólo le queda pulsar sobre uno de ellos y ver su información.

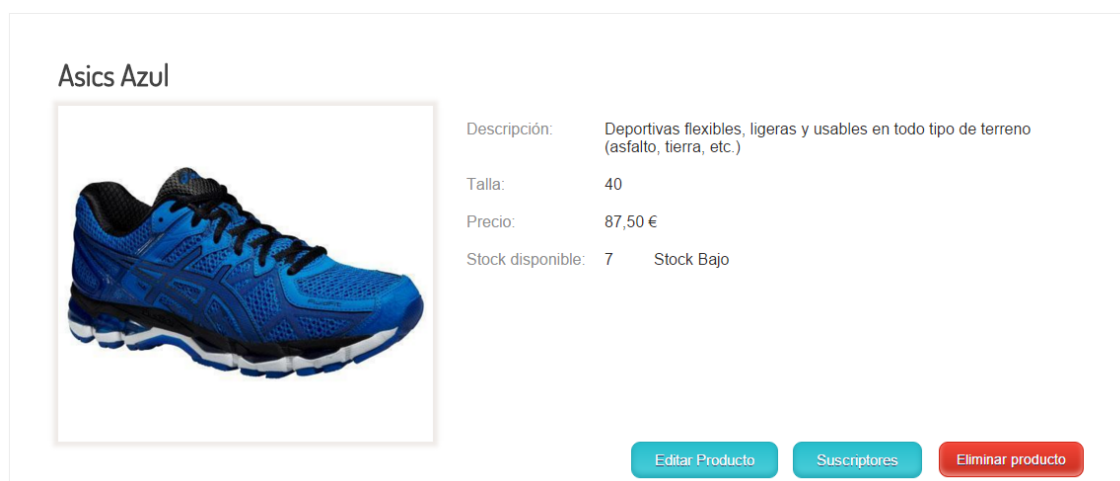
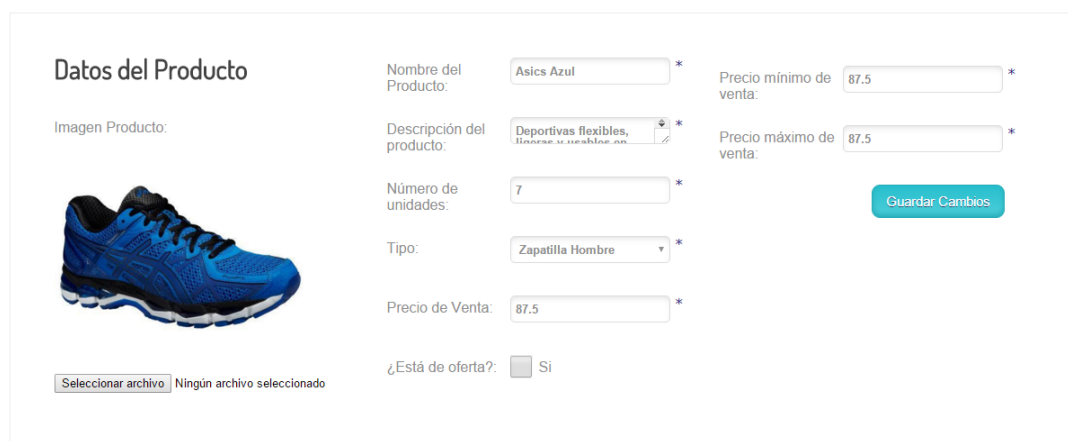


Ilustración 77 - Producto Proveedor


## Editar un producto

Para la edición de un producto, se puede acceder de dos maneras. La primera de ellas se puede ver en la sección “Productos Denegados” y la segunda en la sección “Acceder Producto”. Tras pulsar en ambas páginas sobre el botón “Editar Producto” se desplegará la página con los datos cargados del producto para su posterior actualización. Una vez acabada la edición se deberá pulsar el botón “Guardar Cambios” y se eliminará de la venta para pasar por el visto bueno del administrador.



**Datos del Producto**

Imagen Producto:



[Seleccionar archivo](#) Ningún archivo seleccionado

Nombre del Producto:  \*

Descripción del producto:  \*

Número de unidades:  \*

Tipo:  \*

Precio de Venta:  \*

¿Está de oferta?: ☐ Si

Precio mínimo de venta:  \*

Precio máximo de venta:  \*

[Guardar Cambios](#)

Ilustración 78 - Editar Producto

## Acceder a suscriptores

El proveedor podrá ver que suscriptores tienen sus productos, para ello deberá ir a la sección “Acceder Producto” y pulsar el botón “Suscriptores”. La página desplegada proporciona información sobre el tipo de clientes a los que les gustan los productos del proveedor.

Nombre	Descripción	Dirección	País	Mail	Teléfono
Roberto	Usuario muy activo del running	C/Gainza	Spain	rober93c@gmail.com	654782139






[Anterior](#) [Siguiente](#)

Ilustración 79 - Suscriptor a un producto

## Facturación

Para obtener información sobre la venta de sus productos, el proveedor puede acceder al menú y pulsar sobre el botón “Facturación”. La página que se despliega da información sobre las comisiones del banco y el beneficio que le han reportado sus productos.



Producto	Precio	Fecha	Comisiones	Beneficio
	84,90 €	Sat Feb 06 00:00:00 CET 2016	0,85 €	84,05 €
	84,90 €	Sat Feb 06 00:00:00 CET 2016	0,85 €	84,05 €
	261,89 €	Sat Feb 06 00:00:00 CET 2016	2,62 €	259,27 €
	261,89 €	Sat Feb 06 00:00:00 CET 2016	2,62 €	259,27 €
	261,89 €	Sat Feb 06 00:00:00 CET 2016	2,62 €	259,27 €

Anterior
Siguiente

Ilustración 80 - Facturación Proveedor

## Administrador

A continuación se detallan las acciones que puede realizar el administrador.

### Inicio de sesión

El administrador también debe iniciar sesión antes de entrar en el sistema, para cerrar sesión debe ir a ver la sección “Cierre de sesión”.

El administrador conoce su email y contraseña, ya que es el que maneja toda la aplicación, desde la base de datos hasta la propia web.

En la pantalla desplegada, se debe introducir el email y la contraseña. Una vez introducidos los datos se debe pulsar el botón “Entrar”. Una comprobación rápida para saber si entró al sistema es ver si la parte de arriba cambió.

### Iniciar Sesión

Email

Password

Entrar

### Zona de administración

Bienvenido a la zona de administración de Goal To Run. Desde aquí podrá gestionar de forma completa el sitio web así como los usuarios, proveedores y productos disponibles. Si no es usted administrador, regrese a la página de inicio.

Ilustración 81 - Inicio de Sesión Administrador

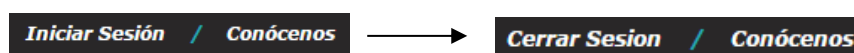


Ilustración 82 - Cambio en la interfaz al iniciar de sesión del Administrador

## Cierre de sesión

Acceder a la parte de arriba de la interfaz y pulsar sobre el botón “Cerrar sesión”, una vez pulsado el usuario habrá abandonado la sesión.

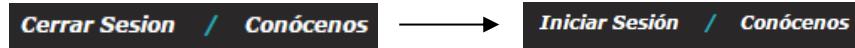


Ilustración 83 - Cambio en la interfaz al cerrar sesión en Administrador

## Validar los productos

El administrador puede acceder a la validación de productos pulsando sobre el botón del menú “Validar Productos”. Una vez dentro, podrá aceptarlos si reúnen los requisitos esenciales que él disponga o denegarlos si no cumplen con su criterio. Para denegar un producto debe ir a la sección “Denegar Producto”.

Inicio > Validar Producto(s)

Imagen	Nombre	Descripción	Unidades	Tipo	Talla	Precio	Oferta	Editar	Eliminar
	Asics Amarillas	Deportivas para todo tipo de terrenos, suaves y cómodas.	9	ZapatillaH	42	82.50€	No	<a href="#">Aceptar</a>	<a href="#">Denegar</a>

[Anterior](#) [Siguiente](#)


Ilustración 84 - Validación de productos

## Denegar los productos

Cuando el administrador desee denegar un producto debido a que no cumple los requisitos que él ha impuesto, debe pulsar sobre el botón “Denegar” de la sección “Validar Productos”. Accederá a una página donde debe darle la explicación al proveedor de porque denegó su producto, para que este pueda modificarlo. Cuando rellene la explicación deberá pulsar el botón de “Enviar” para que le llegue por correo electrónico al proveedor.

### Información del Producto

Imagen:



Nombre:

Descripción:

Precio:

Precio Mínimo:

Precio Máximo:

Tipo:

Talla:

¿Está en Oferta?:

### Explicación de la denegación

Ilustración 85 - Denegar producto de un proveedor

## Ver clientes

Para poder acceder a los clientes que tiene la web, debe pulsar el botón del menú “Clientes”. En esa página se le ofrecerá al administrador toda la información acerca de los clientes. Puede editarlos (Ver sección “Editar cliente”), se le ofrece esta opción si algún cliente hace un mal uso de la web, pudiendo así bloquearlos. O también puede eliminarlos, si quieren darse de baja o cualquier motivo que le lleve al administrador pulsar esta opción. Pero son opciones que se ofrecen para garantizar la seguridad y el buen mantenimiento de la aplicación.

Inicio > Clientes

Nombre	Apellido	Calle	Ciudad	País	CP	Mail	Teléfono	Editar	Eliminar
Jorge	Pedraza	C/Lopez	Madrid	Spain	28054	jorge@mail.com	654789321	<input type="button" value="Editar"/>	<input type="button" value="Eliminar"/>
Maria	Sanz	C/Marcudo	Madrid	Spain	25412	maria@mail.com	654789321	<input type="button" value="Editar"/>	<input type="button" value="Eliminar"/>
Roberto	Canizares	C/Gainza	Madrid	Spain	28041	rober93c@gmail.com	654782139	<input type="button" value="Editar"/>	<input type="button" value="Eliminar"/>
								<input type="button" value="Anterior"/>	<input type="button" value="Siguiente"/>


Ilustración 86 - Ver clientes en la zona de administración

## Editar cliente

Para acceder a la edición del perfil del cliente, el administrador deberá acceder primero a ver todos los clientes (Ver sección “Ver clientes”), una vez dentro deberá pulsar el botón “Editar” sobre el cliente que desee. Se desplegará una nueva página con todos sus datos, se podrá

cambiar todos aquellos datos que se deseen. Una vez que se concluye con el proceso de modificación los cambios se deberán guardar de nuevo en el sistema, para ello se deberá pulsar el botón "Guardar Cambios".

**Datos de Jorge**

Imagen: 

Seleccionar archivo Ningún archivo seleccionado

Nombre:  \*

País:  \*

Apellidos:  \*

Código Postal:  \*

Descripción:  \*

Email:  \*

Dirección:  \*

Contraseña:  \*

Municipio:  \*

Teléfono:  \*

Ciudad:  \*

Ilustración 87 - Edición de los clientes en la zona de administración

## Ver proveedores

Para poder acceder a los proveedores que tiene la web, debe pulsar el botón del menú "Proveedores". En esa página se le ofrecerá al administrador toda la información acerca de ellos. Puede editarlos (Ver sección "Editar Proveedor"), se le ofrece esta opción por si algún proveedor hace un mal uso de la web, pudiendo así bloquearlos. También puede eliminarlos, si quieren darse de baja o cualquier otro motivo que le lleve al administrador pulsar esta opción. Pero son opciones que se ofrecen para garantizar la seguridad y el buen mantenimiento de la aplicación.

Inicio > Proveedores


Nombre	Apellido	Calle	Ciudad	País	CP	Mail	Teléfono	Editar	Eliminar
Roberto	Canizares Sanz	C/Gainza	Madrid	Spain	28041	otrebor430@gmail.com	666548787	<input type="button" value="Editar"/>	<input type="button" value="Eliminar"/>
Pepe	Garcia	C/Beltran	Madrid	Spain	28041	pepe@mail.com	654128793	<input type="button" value="Editar"/>	<input type="button" value="Eliminar"/>

Ilustración 88 - Ver roveedores en la zona de administración


## Editar proveedor

Para acceder a la edición del perfil del proveedor, el administrador deberá acceder primero a ver todos los proveedores (Ver sección "Ver Proveedores"), una vez dentro deberá pulsar el botón "Editar" sobre el proveedor que desee. Se desplegará una nueva página con todos sus datos, se podrá cambiar todos aquellos datos que se deseen. Una vez que se concluye con el

proceso de modificación los cambios se deberán guardar de nuevo en el sistema, para ello se deberá pulsar el botón “Guardar Cambios”.



Datos de Pepe

Imagen: 

Nombre:  \*

Apellidos:  \*

Descripción:  \*

Dirección:  \*

Municipio:  \*

Ciudad:  \*

País:  ▼

Código Postal:  \*

Email:  \*

Contraseña:  \*

Teléfono:  \*

Ningún archivo seleccionado

Ilustración 89 - Edición de los proveedores en la zona de administración

## Conciliación

Para realizar la conciliación, el administrador debe pulsar el botón del menú “Conciliación”, se desplegará una pantalla que le indicará que la conciliación se realizó de manera satisfactoria.

En esta pantalla se realiza tanto la conciliación de la empresa como la de los proveedores.

[Inicio](#) > **Conciliación**

Conciliación Realizada

Ilustración 90 - Conciliación

## Anexo B – Project Abstract

If we analyse the current situation, we can realize that we live in a changing world and evolving. The advance in the Information and Communication Technology (ICT) have done that our daily life is product of a deep change, in which we spend most of our time connected to the Network, wherever we are and with any device. The technology has become into an essential of our day-to-day, arriving almost to the point that sometimes it is totally transparent to us and we are not able to conceive a world without it, every time we can carry more actions out in a virtual world. We live in a "digital" world, as it is the era of *"digitalization of our lives"*.

The e-commerce plays an important role in the life of the *websurfers*. The globalisation in the network has allowed the opening of businesses around the world during the 24 hours per day, per 7 days per week, so why not create a business e-Commerce?

The election to create a business e-commerce as a main subject of this project is due to, greatly, to the use that we do on internet because thanks to it a lot of consumers use their Smartphone, Tablets, Computers, Laptop or their own televisions to get or sell products, creating a new paradigm. The business e-Commerce has an important capital due to the daily multiple operations and electronic purchases that are made. They have had an exponential increase in a few years and a big part of this success is because they have improved the provision of his services and they have got to approach the user to the traditional shop, doing the own website more accessible and intuitive.

Why business e-commerce is more beneficial? Moreover, they are more beneficial because most of the consumers appreciate the comfort of making online purchases or transactions without having to move physically to the store or bank, so in that way, they save time, energy and money.

It pretends, therefore, create and improve a business e-commerce doing use of the best technologies, that we will see afterwards. Of the same way, we looks for strengthen, expand and deepen in the knowledge and the competitions that have gained during the University.

### Motivation

The present project pursues to strengthen the knowledge and the skills that have acquired along the Degree of Computer Engineering. Making connections with both, everything I have learned since I joined in the University in 2011 until now in a single project, it is an arduous and complex task that comports multiple difficulties, but at the same time, it's really motivating because it culminates a wonderful university stage.

Putting up in common all the knowledge, techniques and skills learnt during the following subjects: Programming (Programación), Techniques of Search and Use of Information (Técnicas de Búsqueda y Uso de la Información), Techniques of Oral and Written expression (Técnicas de Expresión Oral y Escrita), Data Structure and Algorithms (Estructura de Datos y Algoritmos), Fundamentals of Business Management (Fundamentos de Gestión Empresarial), Files and Databases (Ficheros y Bases de Datos), Principles of Software Development (Principios de Desarrollo Software), Software Engineering (Ingeniería del software), User Interfaces (Interfaces de Usuario) and Web Computer Technologies (Tecnologías Informáticas para la web). They are the sufficient motivation to develop an ambitious project.

Another reason that promoted me to develop this project has relation with the peak and the popularity that is having the e-commerce in the labour world. We already know that Internet is a source of opportunities but, not everything is valid and to set up a good on-line business you need to get people to know whatever they think and how to improve day to day.

Embarking in this big project so interesting and ambitious involves knowing closely the worries and problems of the e-commerce, and by this way, learns to analyse and correct them in the best possible way. This implication can be a good form to approach me to the business world that there is behind a university degree.

## Goals

The main goal of the project will consist in the development of an application of e-commerce that facilitates the operational of clothes, footwear and sportive accessories shop and the purchase by the Web. The shop will have of a wide catalogue of products, that they are put to disposal of the users by the supplier. The users will be able to access to the information detailed of each product, they could subscribe on it or even, they also could buy it if they wish to. For these last steps, the user must register to initiate session. It has a portal of administration that it will be in charge that the shop work perfectly and to watch the users and the suppliers to avoid an irregular use of the application. The system for the cashing of the products will carry out through a simulation of runways payment.

The main goals that compose this project are:

- ~ Build an attractive, useful, and easy learning eCommerce application.
- ~ Build different portals for the Management of the eCommerce application.
- ~ Simulate the system of charge through the runways payment.
- ~ Make a plan of proofs to check the correct operation of the application.

## Resume

As any software project, the customer generally does not know or does not have an specific idea of how the system works, so it proceeds to make an survey form in which it negotiate the different points that affect it. The first meeting with the customer was taken place on 12th November 2015, and his purpose was to know more details about what he required. After the timely presentations, the customer explained us what he needs. His idea has been collected in the following lines:

*"A web application facilitates the operative of a site/shop available of clothes, footwear and accessories of sport. The final sale shop will contain a catalogue with the clothes, footwear and sportive accessories, at your disposal by a group of supplier. Furthermore, it needs that the cashing system integrates with different runways of payment and it has an available financial management, including the conciliation of the sales. It insists in that the application send messages, two kind of them, first of all a message to suppliers when the product is refused for to inform him why has been denied and second, a message to the users which contain a pdf report with all information about the purchase that the person just made. The development platform for the application must be Java".*

The restrictions imposed by the client are:

- ~ The supplier only will be able to see his products and adjust the prices.
- ~ It is not necessary the implementation of a server.
- ~ The users must register before making a purchase.
- ~ The development platform for the application must be by Java.

The final product will have the following components:

- ~ It needn't to implement a server, after studying and analyse several servers web, we decide by the server web Glassfish because of his stability, to due to it is a container Java EE and to handle the execution of EJB, JMS and, JTA. In addition, it is the implementation of reference for J2EE and is developed by Sun Microsystem.
- ~ The selection of a suitable architecture will be going to sit the bases of the development of a project. In this case, they do not give alternatives of different architectures, the architecture C4 (it is developed later). It was imposed by the tutor of the project looking for consolidate, expand and deepen in the knowledge and competencies that have gained during the university stage. In addition, making more valuable to development the project by making use of the following technologies:



- **JAVA EE:** Java programming platform to develop and execute the application lodged in the server. In combination with a database MySQL can build a faster and effective system.
- **Java Persistence API:** better known by his acronyms JPA, is the API of persistence developed for the platform Java EE. It was chosen as a model of persistence since it is based in POJO's and provides an object/relational mapping.
- **MySQL:** it has free hosting on the web server and support this type of database, it has chosen as the database.
- **JSON:** It has chosen by the simplicity in the exchange of information that produces between customer and server.
- **REST:** architecture for the communication between the customer and the server. Since REST works with the protocol HTTP, the simplicity that possesses in comparison with other methods of communication like CORBA, SOAP or WDSL does that his use was very used in the systems of information of nowadays.

The architecture is based on the technique of modelling diagrams c4 (or the four C: Context, Containers, Components and Classes), since all system must operate around a context and it has to interact with other information technology systems. This system can be modelled as a set of containers, or execution environments, in which software components are executed. At the same time, the components of software could implement like an aggregation of elements, that is, classes or interfaces. Now, the steps of the architecture are shown as follow:

- ~ **Context Diagram:** It uses to represent the limits of the system, that is, it allows differentiating what are the system and his surroundings. Also, it helps to define what the system does and that don't. To develop that, we place the application in the centre of the system, with the three types of entities that we can find:
  - The other systems with which it interacts.
  - The users of the system.
  - The centres of information that you needs.

The focus should be on users and software systems instead of technologies, protocols and other low-level details.

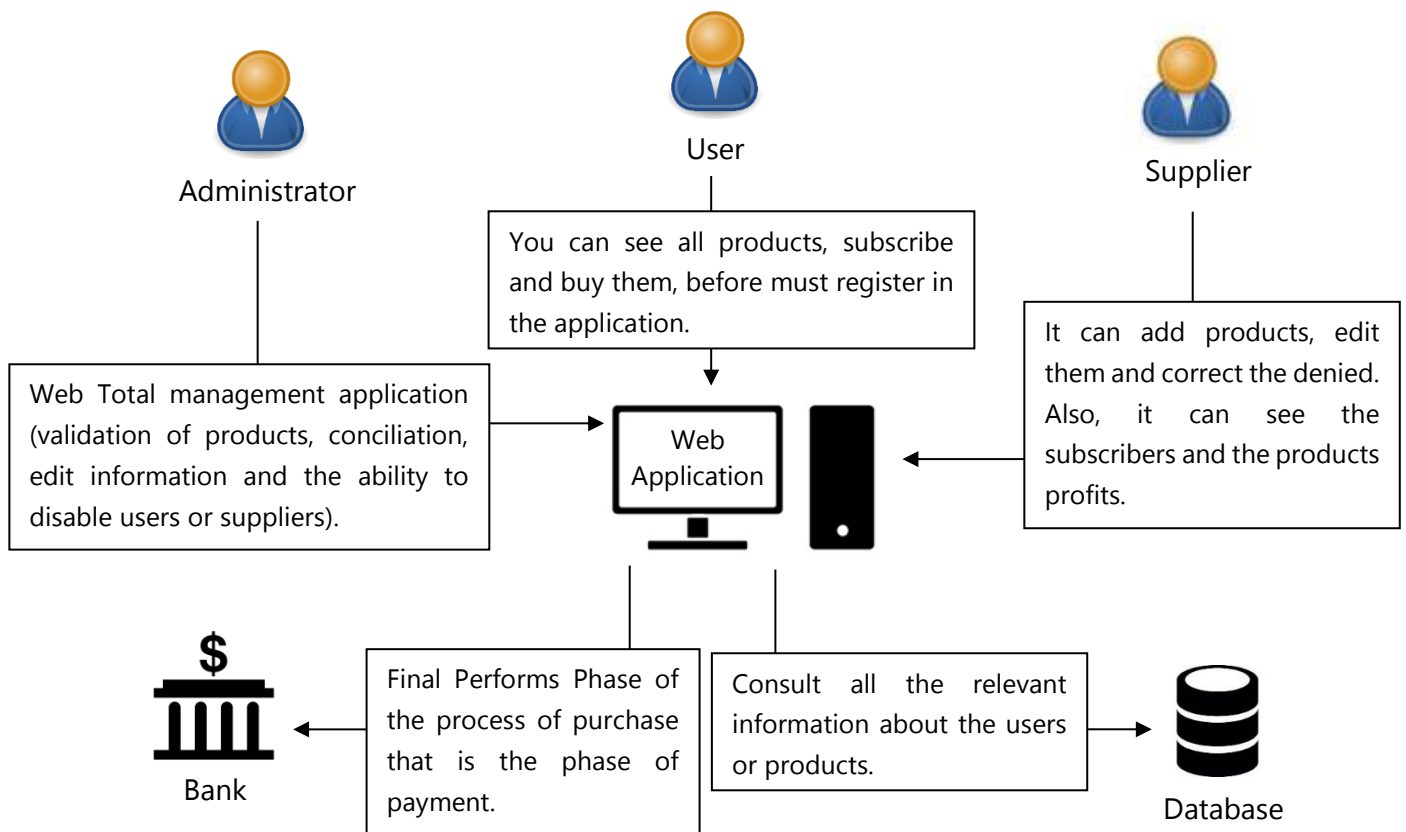


Illustration 91 - Context Diagram

~ **Container Diagram:** If you understand how the system fits into the context diagram, it can be very useful to illustrate the technology options of high level with a diagram of containers. Essentially, a container is any element that can hold data or code. These diagrams shows in high level the architecture of software and how distribute the responsibilities through him. It is a diagram of technology of high level focused for developer's software.

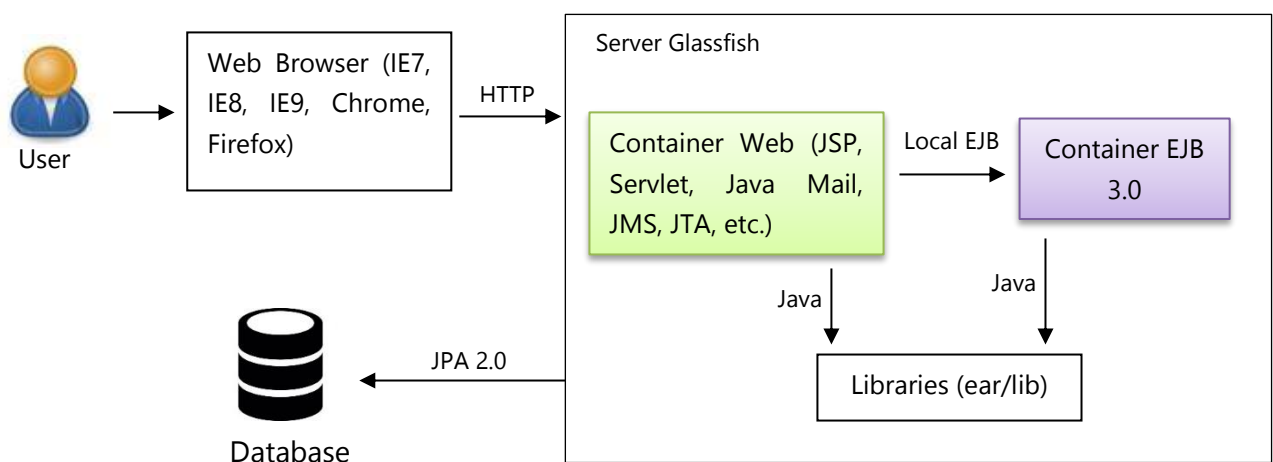


Illustration 92 – User Container Diagram

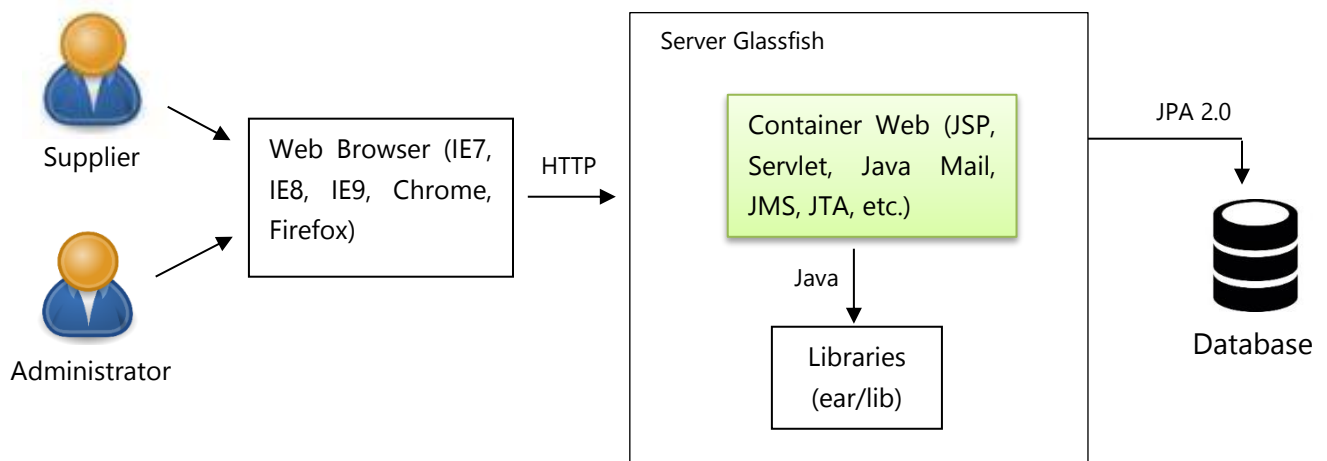


Illustration 93 – Supplier/Administrator Container Diagram

- ~ **Component Diagram:** It decomposes even more all what has been observed in the diagram of containers, that is, each container is due to decompose the main logical components and their interactions. Instead of showing a diagram which can lead to mistakes, in *Chapter IV - System Design* you can find a section called "Code Organization" it appears with full details all components of containers and their interactions.
- ~ **Class Diagram:** This type of diagrams is optional, but sometimes, drawing one or more diagrams of class UML help us to illustrate the details of implementation of a specific component.

## Selected methodology

The choice of methodology has been selected by the project supervisor recommendation; he is working in the part of the I.T. (Information Technologies). His experience, extensive knowledge on this professional area during the degree, caused by the greater use of the methodology in a waterfall, they have been sufficient grounds to develop the project with the Scrum agile methodology.

As definition, *Scrum* is a methodology of management the agile projects in which we apply a regulate way a set of good practices to work collaboratively, in a teamwork, and to obtain the best possible result about this. It is often used by the equipments of development software; in fact, it is the agile methodology more popular.

Scrum relies on three fundamental pillars:

- ~ **Transparency:** any aspects of the process that affect the outcome must be visible and known to everybody involved in the project process. An example of this are the collaborative techniques that are used for communication between the different members of the team.
- ~ **Inspection:** frequent opportunities to review the current state of the work and how the team is using Scrum. This gives insight into current challenges, progress made so far, and creativity the by the team to solve complex problems.
- ~ **Adaptation:** to follow inspection, it is critical to make changes and adapt to the new ideas, information, and current environment of the work. There are also plenty of opportunities to make changes to the work being done, either to the features that are worked on and/or the where that they are being developed.

As another methodology, Scrum also has his model of work. Getting an idea of the model work that I have been done, the following illustration shows it as the perfection. In this illustration, we can observe clearly the different roles that take part during the process and that they will be described in following sections, as well as the different artifacts and events that are also given appointments in a Scrum project.

## The Agile: Scrum Framework at a glance

Información de los ejecutivos,  
el equipo, los implicados,  
los clientes, los usuarios, etc.

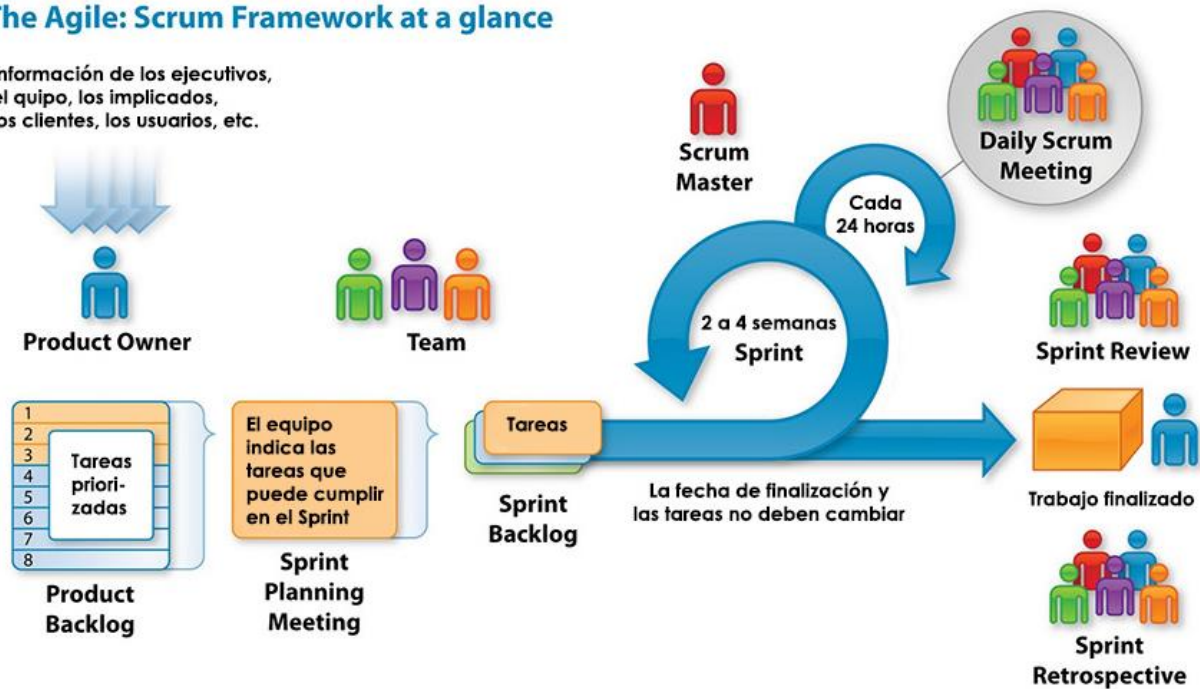


Illustration 94 - Life cycle of Scrum methodology

## Scrum Events

The Scrum events have the goal of create regularity and decrease the need of meetings not defined in the methodology. All the events must have a predefined duration, by the reason of the event can't extend further than the defined duration. The vital events of Scrum are:

- ~ **Sprint:** the heart of Scrum, a time-box of one month or less during which a "Done", useable, and potentially releasable product Increment is created.
- ~ **Sprint Planning Meeting:** the work to be performed in the Sprint is planned in the Sprint Planning Meeting. Is of duration of maximum of four hours for two weeks sprints and eight hours for one month Sprints. Throughout the project, a total of three face-to-face meetings have been made at the end of each Sprint between Product Owner / Scrum Master (Jesús Hernando Corrochano) and Product Owner / Development Team (Roberto Cañizares Sanz).
- ~ **Daily Scrum:** the Daily Scrum Meeting is a 15-minute meeting for the Team, conducted daily to quickly understand the work since the last Daily Scrum Meeting and create a plan for the next 24 hours.
- ~ **Sprint Review:** a Sprint Review is held at the end of every Sprint. During the Sprint Review, a presentation of the increment that is getting released is reviewed. In this meeting, the Scrum Team and the stakeholders collaborate to understand what was

done in the Sprint. The Sprint Review is normally held for two hours for two week sprints and for four hours for one month sprints.

- ~ **Sprint Retrospective:** the Sprint Retrospective is an opportunity for the Scrum Team to inspect itself and create a plan for improvements to be enacted during the next Sprint. This is usually a one hour meeting for two-week duration sprints and a three hour meeting for one month duration Sprints.
- ~ **Backlog Grooming:** meeting with all the Scrum team, including the Product Owner and Scrum Master to review the Product Backlog and add, remove or re-estimate the user stories.

## Scrum Artifacts

Scrum artifacts are still good to represent work due to the complete equipment of Scrum have a transparent vision of the project. The artifacts are:

- ~ **Product Backlog:** the Product Backlog is an ordered list of features that are needed as part of the end product and it is the single source of requirements for any changes to be made to the product. The Product Owner is responsible for the Product Backlog, including its content, availability, and ordering.
- ~ **Sprint Backlog:** the Sprint Backlog is the set of Product Backlog items selected for the Sprint, plus a plan for delivering the product Increment and realizing the Sprint Goal. The Sprint Backlog is a forecast by the Development Team about what functionality will be in the next increment and the work needed to deliver that functionality into a "Done" Increment.
- ~ **Iterative and incremental life cycle:** is the sum of all the Product Backlog items completed during a Sprint and the value of the increments of all previous Sprints. At the end of a Sprint, the new Increment must be "Done," which means it must be in useable condition and meet the Scrum Team's definition of "Done." It must be in useable condition regardless of whether the Product Owner decides to actually release it.

## Conclusions and Personal assessment

Nowadays, we live in a world connected in which the e-commerce is being more and more important. His use increases exponentially and it is already part of the actual society. A lot of people combine the traditional trade with the online trade. The shops that get to integrate these two types of businesses are those that they are going to obtain more advantages than the others.

The Internet creates new ways for citizens to communicate, congregate and share information of a nature social.

One of the main reasons that drive me to choose this project was the promise future that I have in front of me and the immense quantity of new technologies that are beginning to use with the goal of improving more this type of business.

### Positive Conclusions

The most important and positives conclusions are:

- ~ Knowing "stop the feet" to the proposed requirements by the client.
- ~ Adoption of different mentalities to make the project.
- ~ Some knowledge has been perfected.
- ~ Fulfil with the goal to finish the last project degree.
- ~ Methodologies or agile development frameworks are the future in the development of software, when we favour the communication, the agreement and the synergy in the equipments. All this things are called Concurrent Engineering.

### Negative Conclusions

The most important and negatives conclusions are:

- ~ The Scrum methodology is not be able to work equipments of two people
- ~ The big projects shouldn't do only by one person.
- ~ Speed and burned (points history being able to burn a team in a sprint).

## Future Works

They will exposed only those works that they couldn't have done by the reason of the time and that in the case of marketing system, they would be of great value about the actual product that it motivates its sales. Works that have thought are:

- ~ Adaptive Interface
- ~ Register through social networks
- ~ Add comments and assessments to products
- ~ Add advertising
- ~ Improve functionality
- ~ Password recovery
- ~ Update Stock
- ~ Swagger
- ~ Logging system (Trace)